

AD-A148 792

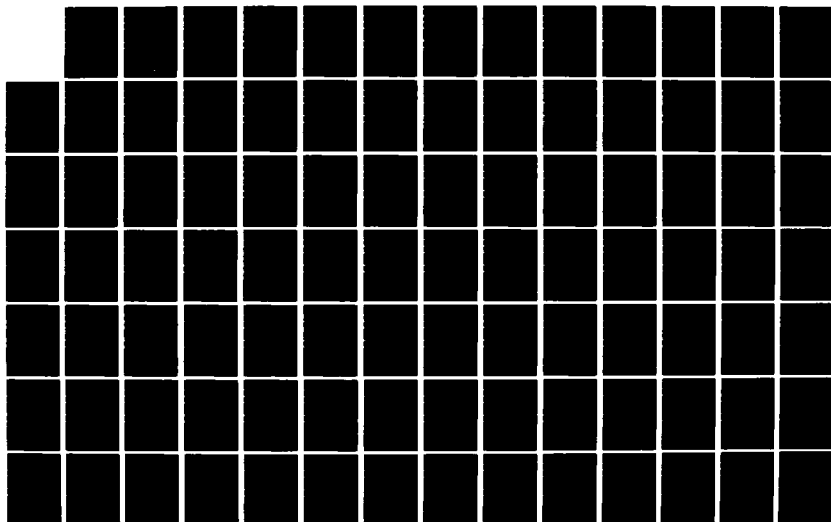
COMPUTER CENTER CDC LIBRARIES/NSRD (SUBPROGRAMS)(U)
DAVID W TAYLOR NAVAL SHIP RESEARCH AND DEVELOPMENT
CENTER BET. D V SOMMER ET AL. JUN 84
DTNSRDC/CMLD-84-12

1/5

UNCLASSIFIED

F/G 9/2

NL



CMLD-84-12

AD-A148 792

COMPUTER CENTER CDC LIBRARIES/NSRDC (SUBPROGRAMS)

DAVID W. TAYLOR NAVAL SHIP RESEARCH AND DEVELOPMENT CENTER

Bethesda, Maryland 20084



COMPUTER CENTER
CDC LIBRARIES/NSRDC (SUBPROGRAMS)

by

David V. Sommer
Sharon E. Good

APPROVED FOR PUBLIC RELEASE: DISTRIBUTION UNLIMITED

Computation, Mathematics and Logistics Department
Departmental Report

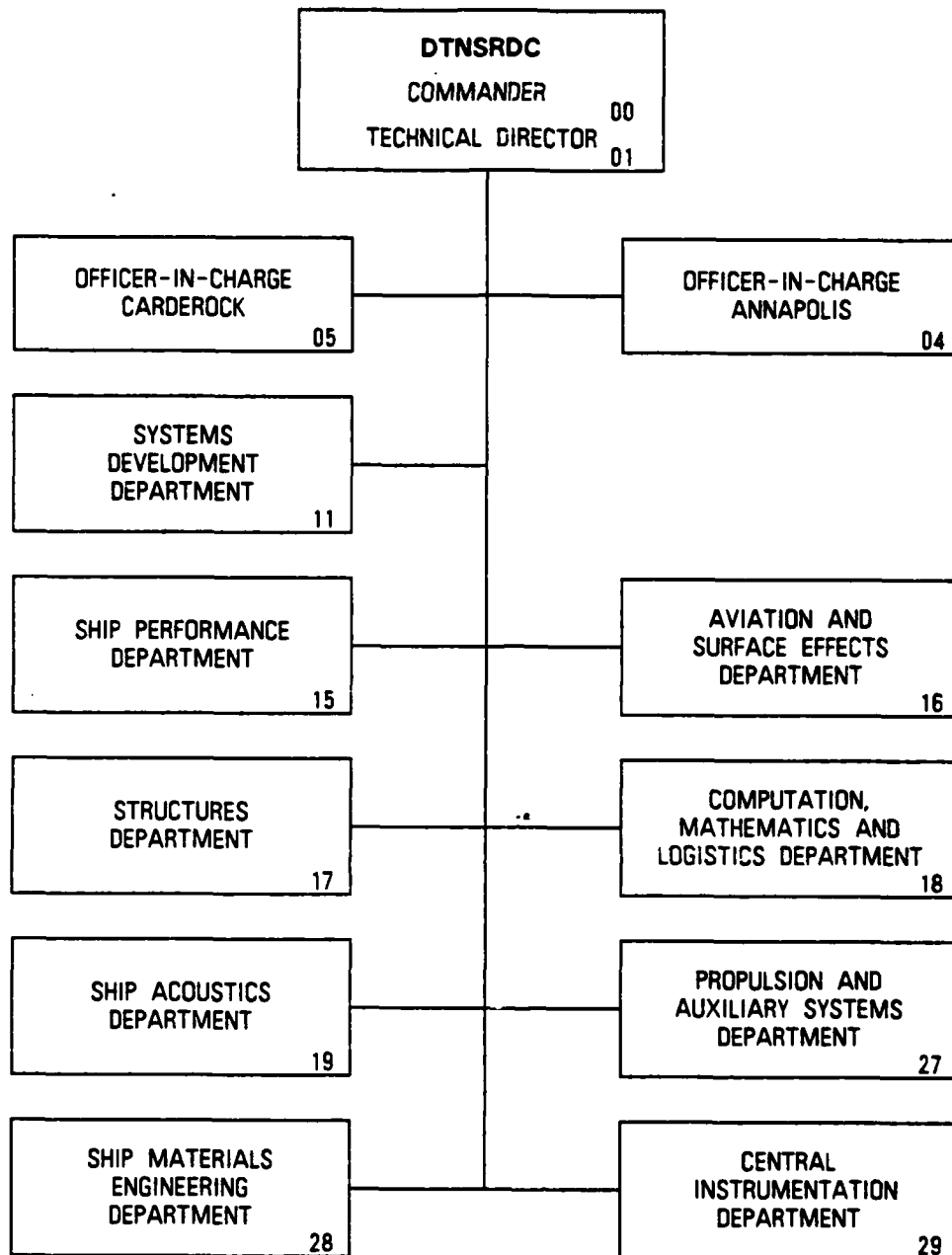
DTIC FILE COPY

June 1984

DTIC
SELECTED
DEC 26 1984
A 22

CMLD-84-12

MAJOR DTNSRDC ORGANIZATIONAL COMPONENTS



REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM								
1. REPORT NUMBER CMLD-84-12	2. GOVT ACCESSION NO. AD A148792	3. RECIPIENT'S CATALOG NUMBER								
4. TITLE (and Subtitle) COMPUTER CENTER CDC LIBRARIES/ NSRDC AND NSRDC5 (SUBPROGRAMS)	5. TYPE OF REPORT & PERIOD COVERED FINAL									
7. AUTHOR(s) DAVID V. SOMMER SHARON E. GOOD	6. PERFORMING ORG. REPORT NUMBER									
9. PERFORMING ORGANIZATION NAME AND ADDRESS DTNSRDC, USER SERVICES, CODE 1892 BETHESDA, MARYLAND 20084	8. CONTRACT OR GRANT NUMBER(s)									
11. CONTROLLING OFFICE NAME AND ADDRESS COMPUTATION, MATHEMATICS & LOGISTICS DEPT. COMPUTER FACILITIES DIVISION (189)	10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS									
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office)	12. REPORT DATE JUNE 1984	13. NUMBER OF PAGES 492								
	15. SECURITY CLASS. (of this report)									
	15a. DECLASSIFICATION/DOWNGRADING SCHEDULE									
16. DISTRIBUTION STATEMENT (of this Report) APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED										
17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)										
18. SUPPLEMENTARY NOTES <i>This document</i>										
19. KEY WORDS (Continue on reverse side if necessary and identify by block number) <table border="0"> <tr> <td>CDC CYBER 170</td> <td>INTERACTIVE PROCESSING</td> </tr> <tr> <td>COMPUTER</td> <td>NOS/BE OPERATING SYSTEM</td> </tr> <tr> <td>CONTROL STATEMENT</td> <td>PROGRAMMING LANGUAGES</td> </tr> <tr> <td>GRAPHICS</td> <td>REMOTE JOB ENTRY</td> </tr> </table>			CDC CYBER 170	INTERACTIVE PROCESSING	COMPUTER	NOS/BE OPERATING SYSTEM	CONTROL STATEMENT	PROGRAMMING LANGUAGES	GRAPHICS	REMOTE JOB ENTRY
CDC CYBER 170	INTERACTIVE PROCESSING									
COMPUTER	NOS/BE OPERATING SYSTEM									
CONTROL STATEMENT	PROGRAMMING LANGUAGES									
GRAPHICS	REMOTE JOB ENTRY									
20. ABSTRACT (Continue on reverse side if necessary and identify by block number) COMPUTER CENTER CDC LIBRARIES/NSRDC (SUBPROGRAMS), CLIB/N, IS A REFERENCE MANUAL WHICH DESCRIBES MOST OF THE SUBPROGRAMS IN LIBRARIES NSRDC AND NSRDC5 ON THE CDC CYBER 170 COMPUTERS AT DTNSRDC. THESE SCIENTIFIC AND UTILITY ROUTINES ARE USED PRIMARILLY WITH FORTRAN PROGRAMS AND MOST ARE WRITTEN IN FORTRAN. CLIB/N LISTS THE ROUTINES BY FUNCTIONAL CATEGORY AND ALPHABETICALLY WITH A DESCRIPTIVE TITLE. ALL AVAILABLE MACHINE-READABLE DOCUMENTS DETAILING THE USE OF THESE ROUTINES ARE INCLUDED.										

19 KEY WORDS (CONTINUED)

HARDWARE

SOFTWARE DOCUMENTATION

DAVID W. TAYLOR
NAVAL SHIP RESEARCH AND DEVELOPMENT CENTER
BETHESDA, MARYLAND 20084

```

*****
*
*
*
*      COMPUTER CENTER
*      CDC LIBRARIES / NSRDC AND NSRDC5
*      (SUBPROGRAMS)
*
*
*****

```

BY
DAVID V SOMMER
USER SERVICES BRANCH
CODE 1892

[illegible]

COMPUTATION, MATHEMATICS AND LOGISTICS DEPARTMENT
DEPARTMENTAL REPORT

CMLD-84-12

TABLE OF CONTENTS

1	INTRODUCTION	1-1
	HOW TO USE THIS MANUAL	1-1
	LIBRARY NSRDC	1-2
	USING LIBRARY NSRDC	1-2
	LIBRARY NSRDC5	1-2
	USING LIBRARY NSRDC5	1-2
	FUNCTIONAL CATEGORIES	1-3
	LIST OF SUBPROGRAMS BY CATEGORY	1-6
	DESCRIPTIVE TITLES (NSRDC)	1-10
	DESCRIPTIVE TITLES (NSRDC5)	1-21
2	PROGRAM DOCUMENTATION (NSRDC)	2-1
	HOW TO PRINT A DOCUMENT	2-1
	<INDIVIDUAL DOCUMENTS	
	ARRANGED ALPHABETICALLY>	2-2 *
3	PROGRAM DOCUMENTATION (NSRDC5)	3-1
	HOW TO PRINT A DOCUMENT	3-1
	<INDIVIDUAL DOCUMENTS	
	ARRANGED ALPHABETICALLY>	3-2 *

* - A LISTING OF THE DOCUMENTS IS NOT INCLUDED IN THIS TABLE OF CONTENTS (SEE PAGE 1-10). AS NEW ROUTINES ARE DEVELOPED, THEY WILL BE INSERTED ALPHABETICALLY INTO THIS DOCUMENT AND MAY BE PRINTED ON THE COMPUTER.

***** INTRODUCTION *****

THE COMPUTER CENTER MAKES AVAILABLE ON THE CDC COMPUTERS, IN ADDITION TO THE NOS/BE OPERATING SYSTEM, A WIDE VARIETY OF BOTH SCIENTIFIC AND UTILITY PROGRAMS, SUBPROGRAMS AND CATALOGUED PROCEDURES. MOST OF THE ROUTINES ARE MAINTAINED IN LIBRARIES ON PERMANENT FILES AND MAY BE INVOKED BY THE APPROPRIATE (LOADER) CONTROL STATEMENTS.

THE CLIB-SERIES OF MANUALS CONSISTS OF THE FOLLOWING, WHICH DESCRIBE THE CONTENTS OF THE VARIOUS CDC CYBER LIBRARIES MAINTAINED BY THE COMPUTER CENTER:

CLIB	- COMPUTER CENTER CDC LIBRARIES	CMLD-84-11
CLIB/N	- COMPUTER CENTER CDC LIBRARIES/NSRDC AND NSRDC5 (SUBPROGRAMS)	CMLD-84-12
CLIB/P	- COMPUTER CENTER CDC LIBRARIES/PROCFIL (PROCEDURES)	CMLD-84-13
CLIB/U	- COMPUTER CENTER CDC LIBRARIES/UTILITY (PROGRAMS)	CMLD-84-14
CLIB/M	- COMPUTER CENTER CDC LIBRARIES/MNSRDC (PROGRAMS)	

THIS MANUAL, CLIB/N, IS A REFERENCE MANUAL WHICH DESCRIBES MOST OF THE SUBPROGRAMS IN LIBRARY 'NSRDC' AND ALL OF THE ROUTINES IN LIBRARY 'NSRDC5'.

*** HOW TO USE THIS MANUAL ***

THE ROUTINES ARE CLASSIFIED IN ONE OR MORE FUNCTIONAL CATEGORIES (SEE PAGE 1-3 FOR A LIST OF CATEGORIES). THEY ARE LISTED, BEGINNING ON PAGE 1-6, UNDER THE VARIOUS CATEGORIES. THE INDIVIDUAL ROUTINES ARE LISTED, WITH DESCRIPTIVE TITLE, BEGINNING ON PAGE 1-9. CHAPTER 2 CONTAINS ALL AVAILABLE MACHINE-READABLE DOCUMENTS DESCRIBING THE USE OF SUBPROGRAMS IN LIBRARY 'NSRDC'. DOCUMENTATION NOT IN CHAPTER 2 MAY BE OBTAINED FROM USER SERVICES, CARDEROCK, BLDG 17, ROOM 100, (202) 227-1907. CHAPTER 3 CONTAINS ALL MACHINE-READABLE DOCUMENTS DESCRIBING THE USE OF SUBPROGRAMS IN LIBRARY 'NSRDC5'.

*** LIBRARY NSRDC ***

'NSRDC' IS A LIBRARY OF DTNSRDC-WRITTEN OR -SUPPORTED SUBPROGRAMS. THESE ROUTINES ARE USED PRIMARILY WITH FTN4, MNF OR RATFOR PROGRAMS AND MOST ARE CODED IN FTN4.

*** USING LIBRARY NSRDC ***

THE FOLLOWING CONTROL STATEMENTS MAY BE USED TO ACCESS 'NSRDC' DURING THE LOADING OF A PROGRAM:

```
...  
FTN4.  -OR-  COBOL.  -OR-  ATTACH,LGO,MYLGO,ID=XXXX.  
ATTACH,NSRDC.  
LDSET,LIB=NSRDC.  -OR-  LIBRARY,NSRDC.  
LGO.  
...
```

*** LIBRARY NSRDC5 ***

'NSRDC5' IS A LIBRARY OF DTNSRDC-WRITTEN OR -SUPPORTED SUBPROGRAMS. THESE ROUTINES ARE CODED IN FTN5 AND USE FEATURES UNIQUE TO ANSI STANDARD FORTRAN 77. THEY MAY NOT BE CALLED BY ROUTINES WRITTEN IN FTN4, MNF OR RATFOR, OR ANY OTHER HIGH-LEVEL LANGUAGE (SUCH AS COBOL OR COBOL5) WHICH DOES NOT SUPPORT FORTRAN 77 FEATURES.

*** USING LIBRARY NSRDC5 ***

THE FOLLOWING CONTROL STATEMENTS MAY BE USED TO ACCESS 'NSRDC5' DURING THE LOADING OF A PROGRAM:

```
...  
FTN5.  -OR-  ATTACH,LGO,MYLGO,ID=XXXX.  
ATTACH,NSRDC5.  
LDSET,LIB=NSRDC5.  -OR-  LIBRARY,NSRDC5.  
LGO.  
...
```

NOTE THAT WHEN BOTH NSRDC AND NSRDC5 ARE REQUIRED, NSRDC5 SHOULD BE MADE AVAILABLE TO THE LOADER FIRST: LDSET,LIB=NSRDC5/NSRDC.

*** FUNCTIONAL CATEGORIES ***

THE FOLLOWING FUNCTIONAL CATEGORIES ARE USED AT DTNSRDC. THOSE CATEGORIES PRECEDED BY AN ASTERISK (*) ARE LOCAL DTNSRDC CATEGORIES. THE OTHER ARE FROM THE VIM (CDC USERS GROUP) LIST.

- A0 ARITHMETIC ROUTINES
- A1 REAL NUMBERS
- A2 COMPLEX NUMBERS
- A3 DECIMAL
- A4 I/O ROUTINES

- B0 ELEMENTARY FUNCTIONS
- B1 TRIGONOMETRIC
- B2 HYPERBOLIC
- B3 EXPONENTIAL AND LOGARITHMIC
- B4 ROOTS AND POWERS

- C0 POLYNOMIALS AND SPECIAL FUNCTIONS
- C1 EVALUATION OF POLYNOMIALS
- C2 ROOTS OF POLYNOMIALS
- C3 EVALUATION OF SPECIAL FUNCTIONS (NON-STATISTICAL)
- C4 SIMULTANEOUS NON-LINEAR ALGEBRAIC EQUATIONS
- C5 SIMULTANEOUS TRANSCENDENTAL EQUATIONS
- * C6 ROOTS OF FUNCTIONS

- D0 OPERATIONS ON FUNCTIONS AND SOLUTIONS OF DIFFERENTIAL EQUATIONS
- D1 NUMERICAL INTEGRATION
- D2 NUMERICAL SOLUTIONS OF ORDINARY DIFFERENTIAL EQUATIONS
- D3. NUMERICAL SOLUTIONS OF PARTIAL DIFFERENTIAL EQUATIONS
- D4 NUMERICAL DIFFERENTIATION

- E0 INTERPOLATION AND APPROXIMATIONS
- E1 TABLE LOOK-UP AND INTERPOLATION
- E2 CURVE FITTING
- E3 SMOOTHING
- E4 MINIMIZING OR MAXIMIZING A FUNCTION

- F0 OPERATIONS ON MATRICES, VECTORS & SIMULTANEOUS LINEAR EQUATIONS
- F1 VECTOR AND MATRIX OPERATIONS
- F2 EIGENVALUES AND EIGENVECTORS
- F3 DETERMINANTS
- F4 SIMULTANEOUS LINEAR EQUATIONS

- G0 STATISTICAL ANALYSIS AND PROBABILITY
- G1 DATA REDUCTION (COMMON STATISTICAL PARAMETERS)
- G2 CORRELATION AND REGRESSION ANALYSIS
- G3 SEQUENTIAL ANALYSIS
- G4 ANALYSIS OF VARIANCE
- G5 TIME SERIES
- G6 SPECIAL FUNCTIONS (INCLUDES RANDOM NUMBERS AND PDF'S)
- * G7 MULTIVARIATE ANALYSIS AND SCALE STATISTICS
- * G8 NON-PARAMETRIC METHODS AND STATISTICAL TESTS
- * G9 STATISTICAL INFERENCE

H0 OPERATIONS RESEARCH TECHNIQUES, SIMULATION & MANAGEMENT SCIENCE
H1 LINEAR PROGRAMMING
H2 NON-LINEAR PROGRAMMING
H3 TRANSPORTATION AND NETWORK CODES
H4 SIMULATION MODELING
H5 SIMULATION MODELS
H6 CRITICAL PATH PROGRAMS
H8 AUXILIARY PROGRAMS
H9 COMBINED

I0 INPUT
I1 BINARY
I2 OCTAL
I3 DECIMAL
I4 BCD (HOLLERITH)
I9 COMPOSITE

J0 OUTPUT
J1 BINARY
J2 OCTAL
J3 DECIMAL
J4 BCD (HOLLERITH)
J5 PLOTTING
J7 ANALOG
J9 COMPOSITE

K0 INTERNAL INFORMATION TRANSFER
K1 EXTERNAL-TO-EXTERNAL
K2 INTERNAL-TO-INTERNAL (RELOCATION)
K3 DISK
K4 TAPE
K5 DIRECT DATA DEVICES

L0 EXECUTIVE ROUTINES
L1 ASSEMBLY
L2 COMPILING
L3 MONITORING
L4 PREPROCESSING
L5 DISASSEMBLY AND DERELATIVIZING
L6 RELATIVIZING
L7 COMPUTER LANGUAGE TRANSLATORS

M0 DATA HANDLING
M1 SORTING
M2 CONVERSION AND/OR SCALING
M3 MERGING
M4 CHARACTER MANIPULATION
M5 SEARCHING, SEEKING, LOCATING
M6 REPORT GENERATORS
M9 COMPOSITE

N0 DEBUGGING
N1 TRACING AND TRAPPING
N2 DUMPING
N3 MEMORY VERIFICATION AND SEARCHING
N4 BREAKPOINT PRINTING

00 SIMULATION OF COMPUTERS AND DATA PROCESSORS (INTERPRETERS)
01 OFF-LINE EQUIPMENT (LISTERS, REPRODUCERS, ETC.)
03 COMPUTERS
04 PSEUDO-COMPUTERS
05 SOFTWARE SIMULATION OF PERIPHERALS
09 COMPOSITE

P0 DIAGNOSTICS (HARDWARE MALFUNCTION)

Q0 SERVICE OR HOUSEKEEPING, PROGRAMMING AIDS
Q1 CLEAR/RESET
Q2 CHECKSUM ACCUMULATION AND CORRECTION
Q3 REWIND, TAPE MARK, LOAD CARDS, LOAD TAPE PROGRAMS, ETC.
Q4 INTERNAL HOUSEKEEPING, SAVE, RESTORE, ETC.
Q5 REPORT GENERATOR SUBROUTINES
Q6 PROGRAM DOCUMENTATION: FLOW CHARTS, DOCUMENT, STANDARDIZATION
Q7 PROGRAM LIBRARY UTILITIES

R0 LOGIC AND SYMBOLIC
R1 FORMAL LOGIC
R2 SYMBOL MANIPULATION
R3 LIST AND STRING PROCESSING
R4 TEXT EDITING

S0 INFORMATION RETRIEVAL

T0 APPLICATIONS AND APPLICATION-ORIENTED PROGRAMS
T1 PHYSICS (INCLUDING NUCLEAR)
T2 CHEMISTRY
T3 OTHER PHYSICAL SCIENCES (GEOLOGY, ASTRONOMY, ETC.)
T4 ENGINEERING
T5 BUSINESS DATA PROCESSING
T6 MANUFACTURING (NON-DATA) PROCESSING AND PROCESS CONTROL
T7 MATHEMATICS AND APPLIED MATHEMATICS
T8 SOCIAL AND BEHAVIORAL SCIENCES AND PSYCHOLOGY
T9 BIOLOGICAL SCIENCES
T10 REGIONAL SCIENCES (GEOGRAPHY, URBAN PLANNING)
T11 COMPUTER ASSISTED INSTRUCTION

U0 LINGUISTICS AND LANGUAGES

V0 GENERAL PURPOSE UTILITY SUBROUTINES
V1 RANDOM NUMBER GENERATORS
V2 COMBINATORIAL GENERATORS: PERMUTATIONS, COMBINATIONS & SUBSETS
* V3 STANDARD AND SPECIAL PROBLEMS

X0 DATA REDUCTION
X1 RE-FORMATTING, DECOMMUTATION, ERROR DIAGNOSIS
X2 EDITING
X3 CALIBRATION
X4 EVALUATION
X5 ANALYSIS (TIME-SERIES ANALYSIS)
X6 SIMULATION (GENERATE TEST DATA FOR DATA REDUCTION SYSTEM)

Y0 INSTALLATION MODIFICATION
Y1 INSTALLATION MODIFICATION LIBRARY
Y2 NEWPL TAPE OF INSTALLATION MODIFICATIONS

Z0 ALL OTHERS

*** LIST OF SUBPROGRAMS BY CATEGORY ***

THE SUBPROGRAMS IN LIBRARIES 'NSRDC' AND 'NSRDC5' ARE LISTED BELOW UNDER THEIR FUNCTIONAL CATEGORIES. ROUTINES FLAGGED WITH AN ASTERISK (*) DO NOT HAVE MACHINE-READABLE DOCUMENTATION. ROUTINES FLAGGED WITH A FIVE (5) ARE IN LIBRARY 'NSRDC5'; ALL OTHERS ARE IN LIBRARY 'NSRDC'. AN ALPHABETICAL LIST OF LIBRARY 'NSRDC', WITH A BRIEF DESCRIPTION OF EACH ROUTINE, BEGINS ON PAGE 1-9; THE ALPHABETICAL LIST OF LIBRARY 'NSRDC5' BEGINS ON PAGE 1-21.

A0 ARITHMETIC ROUTINES

*/ICOMN

A1 REAL NUMBERS

ISUMIT NFill SUMIT

A2 COMPLEX NUMBERS

CMPINV HELP */PSI

B1 TRIGONOMETRIC

*/COTAN

B4 ROOTS AND POWERS

DPROOT PROOT

C1 EVALUATION OF POLYNOMIALS

*/APOWR */HIFAC */POWR1 */PROD2
*/BPOWR */POLDIV */POWR2

C2 ROOTS OF POLYNOMIALS

DPROOT HELP */NROOTS PROOT */QUART

C3 EVALUATION OF SPECIAL FUNCTIONS (NON-STATISTICAL)

*/AI BESSK CELLI */ERROR */LOGGAM
*/BEJYO BESSY */COMBES */EXPINT */PSI
*/BEJY1 BSJ ELLI FRESNEL SNCNDN
BESSI */CBSF */ELLIP GAMCAR
BESSJ */CEI3 */ERF GAMMA

C6 ROOTS OF FUNCTIONS

*/ROOTER

D1 NUMERICAL INTEGRATION

*/FGI QUADG SIMPUN
*/FNOL3 */SIMP */XFIL

D2 NUMERICAL SOLUTIONS OF ORDINARY DIFFERENTIAL EQUATIONS

*/FNOL3 KUTMER

E1 TABLE LOOK-UP AND INTERPOLATION
*/CRDTAB DISCOT */FRMRAN */FRMRA2

E2 CURVE FITTING
FFT */GMHAS OPLSA RFFT */SPLFIT
FFT5 */LSQSUB POLYN RFSN */SQFIT

E3 SMOOTHING
*/SMOOTH

E4 MINIMIZING OR MAXIMIZING A FUNCTION
*/MINMAX

F1 VECTOR AND MATRIX OPERATIONS
MATINS */MATRIX

F2 EIGENVALUES AND EIGENVECTORS
*/MATRIX */VARAH1 */VARAH2

F3 DETERMINANTS
GAUSS MATINS

F4 SIMULTANEOUS LINEAR EQUATIONS
*/BMAM CMPINV */MAM MATINS
CGAUSS GAUSS */MAM200 */MATRIX

G0 STATISTICAL ANALYSIS AND PROBABILITY
*/ACP */DOV */SOV */TOV

G1 DATA REDUCTION (COMMON STATISTICAL PARAMETERS)
*/BDS */CMR */STUTEE

G2 CORRELATION AND REGRESSION ANALYSIS
*/ASA */MRA */SR1 */SR3
*/CMR */PCA */SR2

G4 ANALYSIS OF VARIANCE
*/ANOVA1 */ANOVA2 */AOV

G5 TIME SERIES
*/ASA

G6 SPECIAL FUNCTIONS (INCLUDES RANDOM NUMBERS AND PDF'S)
IAOC IDAYWEK */RANNUM

G8 NON-PARAMETRIC METHODS AND STATISTICAL TESTS
*/RSO

I0 INPUT
*/FASTIN

I2 OCTAL
OFMTDE OFMTV

I3	DECIMAL				
	*/CRDTAB				
I4	BCD (HOLLERITH)				
	*/ICOM	*/ICOMN	IFMTV		
J2	OCTAL				
	PRTFL				
J4	BCD (HOLLERITH)				
	BANR	5/BANR6	LINE6	5/PRTYM	
	5/BANR	*/ICOM	LINE8	5/TTYMSG	
	BANR6	*/ICOMN	PRTIME		
J5	PLOTTING				
	*/PLOTMY	PLOTPR	*/PLOTXY		
K2	INTERNAL-TO-INTERNAL (RELOCATION)				
	CMMOVEF	GETRA	MOVEIT	RCPA	
	GETDABA	MFETCH	MSET	SWAP	
L0	EXECUTIVE ROUTINES				
	5/TTYMSG				
L3	MONITORING				
	GETCCL				
M0	DATA HANDLING				
	COMPSTR	EQU60	MASKIT		
M1	SORTING				
	ASORT	5/CSORT	5/CSORT2	QSORT1	SSORTI
	ASORTMV	5/CSORTD	ISSORT	SSORT	SSORTL
	5/CSHUFL	5/CSORTN	QSORT	SSORTF	SSORT3
M2	CONVERSION AND/OR SCALING				
	5/CHIN	5/CVINCH	HEX3	JGDATE	5/NEWDAT
	5/CVCHIN	DATCNV	5/HMS2S	JULIAN	5/S2HMS
	5/CVCHOL	DATFMT	IHMS	MONTH	UNHEX3
	5/CVHOCH	GETHOURL	IROMAN	NEWDAT	WEKDAY
M4	CHARACTER MANIPULATION				
	ADJL	5/CHIN	GETCHR	OMRONI	SKWEZL
	ADJR	CHNGSEQ	*/GETPRM	PARGET	SKWEZR
	ASCADD	CONTRCT	5/GETSTR	PUTCHA	TEKTRI
	ASCADM	5/CVCHIN	IBUNP	PUTCHR	TRAILBZ
	ASCBSX	5/CVCHOL	IPAKLFT	REPLAC	5/TRANS
	ASCGET	5/CVHOCH	ISTAPE	REPLACM	VALDAT
	ASCI1	5/CVINCH	5/ITRANS	REPLHI	VFILL
	ASCI11	D630I	LBYT	REPLLO	VT100I
	ASCLEN	EXPAND	5/LEFT	REPLNE	ZBLANK
	ASCPUT	EXPRM	LEFTADJ	5/RIGHT	ZEROFL
	ASHIFT	EXTBIT	MOVCHAR	SBYT	ZEROS
	CENTER	EXTPRM	MOVECM	SEMICO	
	5/CENTER	FBINRD	MOVSTR	SETREW	
	CHFILL	GETCHA	MXGET	SHIFTA	

M5 SEARCHING, SEEKING, LOCATING

AMAXE	FINDWRD	5/GETSTR	5/LASTCH	MINE
AMINE	5/FIRSTCH	IDIGIT	5/LASTCHH	NFILLT
5/CFIND	5/FRSTCH	IFINDCH	LASTWRD	5/NUMER
FINDC	GETCHA	LASTC	5/LSTCH	VALIDT
FINDW	GETCHR	LASTCH	MAXE	

N0 DEBUGGING

ALTIME	5/CMPGFS	5/CMPGSS	PRTIME
5/ALTYM	5/CMPGOS	GETCCL	5/SM5PRNT

N2 DUMPING

5/CMDUMP	DMPCPA	DUMPA	5/DUMPPXPK
DMPA	5/DMPCPA	DUMPFL	RECOVRD

O1 OFF-LINE EQUIPMENT (LISTERS, REPRODUCERS, ETC.)
WARNING

Q0 SERVICE OR HOUSEKEEPING, PROGRAMMING AIDS

AC	ELTIME	HERE	LIBSYM	5/PFRC
5/AC	5/ELTYM	IBL	MACHINE	5/PM
ALTIME	FTNRFL	IDID	MEMUSED	PRTFL
5/ALTYM	GETCCL	ISEC	MFRAME	REDUCE
BANR	GETDABA	JOBCM	5/MFRAME	ROUTERC
BANR6	GETFIT	JOBNAME	NUMEXEC	5/ROUTERC
BUFSIZE	GETLFNS	JOBORG	NUMVAR	SKPSTAT
5/CMMERC	GETLGO	LFPFERR	OVLNAME	TIMLEFT
5/CMMUERC	GODROP	LIBBAM	PFRC	

Q3 FILE MANIPULATION

CLUNLD	SKPFIL	UNLOAD
PF	5/TTYOPN	ZSYSEQ

Q4 INTERNAL HOUSEKEEPING, SAVE, RESTORE, ETC.
PRTIME 5/PRTYMR1 FORMAL LOGIC
COUPLE

S0 INFORMATION RETRIEVAL

5/IDID	5/JOBORG
--------	----------

T4 ENGINEERING
*/ARDCFT

V1 RANDOM NUMBER GENERATORS

*/RANNU	RNDMIZ	*/RN1	*/RN2
---------	--------	-------	-------

Z0 ALL OTHERS

DAYONOF	MF2CPU	5/MF2CPU
---------	--------	----------

*** DESCRIPTIVE TITLES (NSRDC) ***

SUBPROGRAMS IN LIBRARY 'NSRDC' ARE LISTED ALPHABETICALLY BELOW.

AC	GET ACCOUNT NUMBER FOR THIS JOB
ACP	ADD CROSS PRODUCT VARIABLES - STATISTICS
ADJL	LEFT ADJUST A LINE OF WORDS LEAVING ONE SPACE BETWEEN WORDS
ADJR	RIGHT ADJUST A LINE OF WORDS LEAVING ONE SPACE BETWEEN WORDS
AI	AIRY FUNCTION INTEGRAL
ALTIME	OBTAIN CPA, CPB, CP, PP, IO AND WALL CLOCK TIMES SINCE START OF JOB (OR INTERCOM SESSION)
AMAXE	FIND MAXIMUM VALUE OF AN ARRAY (ALSO CONTAINS MAXE)
AMINE	FIND MINIMUM VALUE OF AN ARRAY (ALSO CONTAINS MINE)
ANOVA1	ONE-WAY ANALYSIS OF VARIANCE WITH UNEQUAL N
ANOVA2	TWO-WAY ANALYSIS OF VARIANCE WITH EQUAL N
AOV	ANALYSIS OF VARIANCE FROM EQUAL NUMBER OF EQUAL WEIGHT DESIGNS - TOTALS, DEVIATES, SUMS OF SQUARES, DEGREES OF FREEDOM, MEAN SQUARES
APOWR	EXPONENTIATION OF POWER SERIES - ONE VARIABLE
ARDCFT	PROPERTIES OF U.S. STANDARD ATMOSPHERE (1962)
ASA	AUTOCORRELATION AND SPECTRAL ANALYSIS FROM STATIONARY TIME SERIES, GIVES POWER SPECTRUM, LAGGED SUMS AND PRODUCTS
ASCADD	ADD AN ASCII STRING TO ANOTHER ASCII STRING
ASCADM	ADD AN ASCII STRING TO ANOTHER ASCII STRING MULTIPLE TIMES
ASCBSX	REMOVE BS (BACKSPACE) AND CAN (CTRL-X) FROM A STRING
ASCGET	GET AN ASCII CHARACTER FROM AN ASCII STRING
ASCII	CREATE AN ASCII MESSAGE FROM STRINGS OF ASCII CHARACTERS
ASCI11	INITIALIZE COMMON BLOCK /ASCII/ WITH ASCII CHARACTERS
ASCLN	FIND LENGTH OF AN ASCII STRING
ASCPUT	ADD AN ASCII CHARACTER TO AN ASCII STRING
ASCTXT	CONVERT DISPLAY CODE STRING TO ASCII STRING

ASHIFT	SHIFT EACH WORD OF AN ARRAY
ASORT	FTN ALPHANUMERIC SORT
ASORTMV	SORT 2-DIMENSIONAL ARRAY USING A FAST ARRAY MOVING SUBROUTINE
BANR	PRINT A BANNER (LETTERS ARE 10 LINES HIGH, LINES ARE 110 CHARACTERS LONG)
BANR6	PRINT A BANNER (LETTERS ARE 6 LINES HIGH, LINES ARE 80 CHARACTERS LONG)
BDS	BASIC DESCRIPTIVE STATISTICS - MEAN, SECOND, THIRD, FOURTH MOMENTS, VARIANCE, STANDARD DEVIATION, SKEWNESS, KURTOSIS
BEJY0	ZERO-ORDER BESSEL FUNCTIONS FOR REAL ARGUMENTS
BEJY1	FIRST ORDER BESSEL FUNCTIONS FOR REAL ARGUMENTS
BESSI	MODIFIED BESSEL FUNCTION OF THE FIRST KIND
BESSJ	BESSEL FUNCTION OF THE FIRST KIND
BESSK	MODIFIED BESSEL FUNCTION OF THE SECOND KIND
BESSY	BESSEL FUNCTION OF THE SECOND KIND
BMAM	SOLVE SYSTEM $AX=B$ FOR BANDED SYMMETRIC MATRICES
BPOWR	EXPONENTIATION OF POWER SERIES IN TWO VARIABLES
BSJ	SPHERICAL BESSEL FUNCTION
BUFSIZE	PRINT MESSAGE IN DAYFILE FOR EACH FILE SPECIFIED INDICATING BUFFER SIZE AND WHETHER BUFFER IS CURRENTLY ALLOCATED
CBSF	COMPLEX BESSEL FUNCTION FOR LARGE ARGUMENT
CCALL	EXIT PROGRAM AND EXECUTE ONE OR MORE CONTROL CARD
CEI3	COMPLETE ELLIPTIC INTEGRAL OF THE THIRD KIND
CELLI	COMPLETE AND INCOMPLETE ELLIPTIC INTEGRALS OF THE FIRST AND SECOND KIND
CENTER	CENTER A CHARACTER STRING WITHIN AN OUTPUT FIELD
CFILL	FILL AREA WITH ALTERNATING FIELDS OF SPECIFIED CHARACTER AND BLANKS
CGAUSS	COMPLEX SOLUTION OF SIMULTANEOUS EQUATIONS AND DETERMINANT BY ITERATIVE GAUSSIAN ELIMINATION
CHFILL	FILL (PORTION OF) AN ARRAY WITH A CHARACTER
CHNGSEQ	ALLOW COBOL4 USER TO DEFINE A COLLATING SEQUENCE

CLUNLD	CLOSE AND UNLOAD A FILE
CMPINV	COMPLEX MATRIX INVERSION
CMR	CORRELATION MATRIX WITH OPTIONAL MEAN AND STANDARD DEVIATION
COMBES	BESSEL FUNCTIONS FOR COMPLEX ARGUMENT AND ORDER
COMPSTR	COMPARE TWO CHARACTER STRINGS
CONTRCT	SQUEEZE ARRAY OF 1R-FORMAT CHARACTERS TO LEFT (SEE EXPAND)
COTAN	COTANGENT FUNCTION
COUPLE	LOGICALLY CONNECT TWO WORDS
CRDTAB	READ TABLES FOR FRMRAN AND FRMRA2 INTERPOLATION
DATCNV	CONVERT DATE FORMATS (USES INTEGERS)
DATFMT	CONVERT DATE FORMATS (USES CHARACTER STRINGS)
DAYONOF	PACKAGE OF SIX SUBROUTINES TO MANIPULATE THE DAYFILE SETTING SETTINGS
DISCOT	SINGLE OR DOUBLE INTERPOLATION
DMPA	CALLABLE OCTAL AND CHARACTER DUMP OF SPECIFIED PORTION OF USER'S FIELD LENGTH (FL) (BY ACTUAL LOCATION) (NO HEADINGS ARE PROVIDED)
DMPCPA	DUMP JOB CONTROL POINT AREA
DOV	DELETION OF VARIABLES - STATISTICS
DPROOT	FIND ALL ROOTS OF A REAL DOUBLE PRECISION POLYNOMIAL
DUMPA	GIVE OCTAL AND CHARACTER DUMP OF USER-SPECIFIED AREA
DUMPCPA	EXPANDED DUMP OF JOB CONTROL POINT AREA

DUMPFL	CALLABLE OCTAL AND CHARACTER DUMP OF SPECIFIED PORTION OF USER'S FIELD LENGTH (FL) (BY ACTUAL LOCATION)
D630I	INITIALIZE COMMON BLOCK /D630/ WITH ASCII CONTROL CODES FOR DIABLO 630 TERMINALS
ELLI	ELLIPTIC INTEGRAL
ELLIP	ELLIPTIC INTEGRAL
ELTIME	OBTAIN CPA, CPB, CP, PP, IO AND WALL CLOCK TIMES SINCE LAST CALL TO ELTIME
EQU60	LOGICAL COMPARE OF TWO ARRAYS
ERROR	ERROR FUNCTION
EXPAND	EXPAND CHARACTER STRING INTO ARRAY OF 1R-FORMAT WORDS (SEE CONTRCT)
EXPINT	EXPONENTIAL INTEGRAL
EXPRM	EXTRACT NEXT PARAMETER FROM EXECUTE CARD
EXTBIT	EXTRACT BITS FROM A WORD
EXTPRM	EXTRACT NEXT PARAMETER FROM USER-SUPPLIED PARAMETER STRING
FASTIN	READ AND UNPACK DATA PREPARED ON THE XDS-910 A/D CONVERSION SYSTEM
FBINRD	UNPACK AN INPUT ARRAY (N BITS PER INPUT CHARACTER INTO CDC WORD)
FFT	FAST FOURIER TRANSFORM FOR COMPLEX TABULATED FUNCTION
FFT5	FAST FOURIER TRANSFORM
FGI	FORTRAN GAUSSIAN INTEGRATION
FINDC	FIND PRESENCE OR ABSENCE OF SPECIFIED CHARACTER IN AN ARRAY (USER SPECIFIES RELATIONAL OPERAND)
FINDW	FIND PRESENCE OR ABSENCE OF SPECIFIED WORD IN AN ARRAY (USER SPECIFIES RELATIONAL OPERAND)
FINDWRD	FIND SPECIFIED WORD IN AN ARRAY
FNOL3	INTEGRATE SYSTEM OF ORDINARY DIFFERENTIAL EQUATIONS
FRESNEL	EVALUATE FRESNEL INTEGRALS
FRMRAN	LINEAR TABLE INTERPOLATION (ONE OR TWO INDEPENDENT VARIABLES)
FRMRA2	LINEAR TABLE INTERPOLATION (MULTIPLE INDEPENDENT VARIABLES)
FTNRFL	GET/SET CORE SIZE

GAMCAR	COMPLEX GAMMA FUNCTION OF A COMPLEX ARGUMENT HAVING POSITIVE REAL PART
GAMMA	INCOMPLETE OR COMPLETE GAMMA FUNCTION
GAUSS	SIMULTANEOUS EQUATION SOLUTION WITH DETERMINANT BY ITERATIVE GAUSSIAN ELIMINATION
GETCCL	GET CCL FIELDS (REGISTERS AND FLAGS)
GETCHA	EXTRACT CHARACTER FROM SPECIFIED POSITION IN AN ARRAY
GETCHR	EXTRACT CHARACTER FROM SPECIFIED POSITION IN A WORD
GETDABA	GET DYNAMIC AREA BASE ADDRESS AND DETERMINE IF CMM IS ACTIVE
GETFIT	GET SPECIFIED FIT ADDRESS
GETHOUR	FOR A SPECIFIED PERIOD OF TIME (UP TO 2 HR 59 MIN 59 SEC) DETERMINE WHICH HOUR IS OCCUPIED THE LONGEST
GETLFNS	GET ACTUAL LOCAL FILE NAMES (FOR FTN)
GETLGO	EXTRACT FIRST 10 CHARACTERS OF ALL EXECUTE CARD PARAMETERS
GETRA	GET PROGRAM COMMUNICATION REGION (RA+0 THRU RA+77B)
GMHAS	HARMONIC ANALYSIS
GODROP	ISSUE USER-SPECIFIED GO/DROP MESSAGE
HELP	COMPLEX ZEROES OF REAL OR COMPLEX POLYNOMIAL
HERE	GET TERMINAL ID FOR THIS JOB
HEX3	SQUEEZE 3-CHARACTER HEX INTO 12 BITS
HIFAC	HIGHEST COMMON FACTOR OF TWO POLYNOMIALS
IAOC	COUNT ONE-BITS IN SPECIFIED WORD
IBL	CALCULATE BEST BLOCK LENGTH (MIN TIME REQ'D FOR RANDOM ACCESS AND MINIMUM BUFFER SIZE) FOR INDEX SEQUENTIAL FILES
IBUNP	UNPACK 12-BIT BYTES FROM ARRAY
IDAYWEK	FUNCTION TO DETERMINE THE DAY OF THE WEEK FOR ANY DATE FROM 10/15/1582 THRU 02/28/4000
IDID	GET USER INITIALS (AND INTERCOM USER ID) FROM CHARGE CARD OR LOGIN
IDIGIT	CHECK FOR DIGITS IN A FIELD WITHIN A WORD
IFINDCH	FIND FIRST OCCURRENCE OF SPECIFIED CHARACTER IN ARRAY

IFMTV FAST I-FORMAT DECODE OF VARIABLE LENGTH INPUT

IHMS CONVERT SECONDS TO ' HH.MM.SS.' (SEE ISEC)

IPAKLFT SQUEEZE LEFT AND REMOVE ZEROS (OOB) AND BLANKS (55B), RETURN
NUMBER OF CHARACTERS

IROMAN CONVERT ROMAN NUMBERS TO INTEGER

ISEC CONVERT HH.MM.SS TO SECONDS (SEE IHMS)

ISITCNF TEST FOR CONNECTED FILE

ISSORT FTN-CALLABLE SHELL SORT FOR INTEGER ARRAYS

ISTAPE GENERATE TAPE NAME 'TAPENN'

ISUMIT SUM ELEMENTS OF INTEGER ARRAY

JGDATE CONVERT ANY GREGORIAN DATE TO A JULIAN DATE AND VICE VERSA
(MULTI-YEAR)

JOBCH GET JOB CARD CH

JOBNAME GET NOS/BE JOB NAME FOR THIS JOB

JOBORG GET JOB ORIGIN (BATCH, INTERCOM, GRAPHICS, MULTI-USER)

JULIAN CONVERT ANY GREGORIAN DATE TO A JULIAN DATE AND VICE VERSA
(SINGLE YEAR)

KUTMER INTEGRATE A SYSTEM OF FIRST-ORDER ORDINARY DIFFERENTIAL
EQUATIONS USING THE KUTTA-MERSON FOURTH-ORDER, SINGLE-STEP
METHOD

LASTCH FIND LAST NON-BLANK CHARACTER IN ARRAY

LASTWRD FIND SUBSCRIPT OF LAST WORD OF ARRAY WHICH CONTAINS A
NON-BLANK

LBYT EXTRACT VARIABLE LENGTH BYTE

LEFTADJ SQUEEZE LEFT AND REMOVE BLANKS AND OOB (USER MAY SUPPLY
TRAILING FILL CHARACTER)

LFPFERR DECODE THE "ERR" CODE FROM FILE MANIPULATION SUBROUTINES PF
AND LF

LIBBAM DUMMY SUBROUTINE TO FORCE LDSET,LIB=BAMLIB

LIBSYM DUMMY SUBROUTINE TO FORCE LDSET,LIB=SYMLIB

LINE6 SET PRINT FILE TO 6 LINES PER INCH

LINE8 SET PRINT FILE TO 8 LINES PER INCH

LOGGAM LOGARITHM OF GAMMA FUNCTION FOR COMPLEX ARGUMENT

LSQSUB GENERAL WEIGHTED LEAST SQUARES FIT

MAM SOLVE SYMMETRIC SYSTEM OF LINEAR EQUATIONS

MAM200 SOLVE 200 SYMMETRIC LINEAR EQUATIONS

MASKIT DYNAMIC MASK GENERATOR

MATINS MATRIX INVERSE WITH SIMULTANEOUS EQUATION SOLUTION AND DETERMINANT

MATRIX MATRIX ALGEBRA - TRANSPOSE, MOVE, SYMMETRIC PRODUCT, EIGEN-VALUE/EIGENVECTOR, PACK SYMMETRIC, UNPACK SYMMETRIC, INVERSE, SOLUTION OF LINEAR EQUATIONS, MULTIPLY, ADD, SUBTRACT, TRANSPOSE MULTIPLY.

MAXE FIND MAXIMUM VALUE OF AN ARRAY (ALSO CONTAINS AMAXE)

MEMUSED PRINT MESSAGE IN DAYFILE GIVING FIELD LENGTH IN USE AT TIME OF CALL TO THIS ROUTINE

MFETCH FETCH A SINGLE WORD FROM USER'S FL (SEE MSET)

MFRAME OBTAIN THE MACHINE AND MAINFRAME RUNNING THE PROGRAM

MF2CPU RETURN CPU NAME CORRESPONDING TO SUPPLIED MAINFRAME NAME

MINE FIND MINIMUM VALUE OF AN ARRAY (ALSO CONTAINS AMINE)

MINMAX GENERALIZED NONLINEAR ITERATOR

MONTH FROM A DATE (MM/DD/YY) FIND THE MONTH AND RETURN FULL SPELLING AND 3- OR 4-CHARACTER ABBREVIATION

MOVCHAR MOVE ONE CHARACTER FROM ONE STRING TO ANOTHER

MOVECM MOVE WORDS FROM ONE AREA IN CORE TO ANOTHER

MOVEIT MOVE AN ARRAY (MOVLEV REPLACEMENT WHICH CALLS MOVECM)

MOVSTR MOVE A STRING OF CHARACTERS FROM ONE ARRAY TO ANOTHER

MRA MULTIPLE REGRESSION ANALYSIS - LEAST SQUARES ESTIMATE OF LINEAR RELATIONSHIPS

MSET SET A SINGLE WORD IN USER'S FL (SEE MFETCH)

MXGET EXTRACT (RIGHT-JUSTIFIED, ZERO-FILLED) 0-10 6-BIT CHARACTERS FROM 60-BIT WORDS

NEWDAT ADD/SUBTRACT SPECIFIED NUMBER OF DAYS TO/FROM A GIVEN DATE

NFILL FILL ELEMENTS 1 THRU N OF AN ARRAY WITH THE VALUES 1 THRU N, RESPECTIVELY

NFILLT TEST AN ARRAY FOR THE PRESENCE OF THE INTEGERS 1 THRU N IN
ELEMENTS 1 THRU N, RESPECTIVELY

NROOTS REAL AND COMPLEX ROOTS OF REAL POLYNOMIAL

NUMEXEC GET NUMBER OF EXECUTE CARD PARAMETERS WHICH WERE USED IN THIS
EXECUTION OF THE PROGRAM

NUMVAR DETERMINE NUMBER OF ARGUMENTS IN CALL TO SUBPROGRAM

OFMTDE FAST O-FORMAT DECODE

OFMTV FAST O-FORMAT DECODE OF VARIABLE LENGTH INPUT

OMRONI INITIALIZE COMMON BLOCK /OMRON/ WITH ASCII CONTROL CODES
FOR OMRON CRT'S

OPLSA ORTHOGONAL POLYNOMIAL LEAST SQUARE APPROXIMATION

OVLNAME GET NAME OF FILE CURRENTLY BEING EXECUTED

PARGET GET ALL PARAMETERS OF USER-SUPPLIED PARAMETER STRING

PCA PRINCIPLE COMPONENT ANALYSIS - EIGENVALUES AND EIGENVECTORS
OF CORRELATION MATRIX, TRANSFORMS NORMALIZED OBSERVATION INTO
ORTHOGONAL COMPONENTS AND CHECKS ACCURACY

PF FORTRAN CALLABLE PERMANENT FILE FUNCTIONS AND AUXILIARY FILE
ACTION REQUESTS

PFR SUPPLY DESCRIPTION OF PERMANENT FILE FUNCTION RETURN CODE

PLOTMY PRINTER PLOT - MULTIPLE CURVES

PLOTPR PRINTER PLOT - MULTIPLE CURVES

PLOTXY PRINTER PLOT - SINGLE CURVE

POLDIV POLYNOMIAL DIVISION

POLYN LEAST SQUARES POLYNOMIAL FIT

POWR1 1 TERM IN EXPONENTIATION OF POWER SERIES - ONE VARIABLE

POWR2 1 TERM IN EXPONENTIATION OF POWER SERIES - TWO VARIABLES

PROD2 1 TERM IN PRODUCT OF POWER SERIES - TWO VARIABLES

PROOT FIND ALL ROOTS OF A REAL PLOYNOMIAL

PRTFL PRINT CURRENT FL (OR PUT INTO DAYFILE)

PRTIME GET AND PRINT CPA, CPB, CP, PP, IO AND WALL CLOCK TIMES SINCE
LAST CALL AND PRINT USER-SUPPLIED MESSAGE

PSI COMPLEX PSI FUNCTION

PUTCHA	INSERT CHARACTER INTO SPECIFIED POSITION IN AN ARRAY
PUTCHR	INSERT CHARACTER INTO SPECIFIED POSITION IN A WORD
QSORT	IN-CORE ASCENDING SORT FOR ARRAYS LARGER THAN 500 WORDS
QSORT1	IN-CORE ASCENDING SORT WITH RE-ORDERING OF ASSOCIATED ARRAY (FOR ARRAYS LARGER THAN 500 WORDS)
QUADG	INTEGRAL BY GAUSS-LEGENDRE 10-POINT QUADRATURE
QUART	REAL OR COMPLEX ROOTS OF QUARTIC
RANNUM	NORMALLY DISTRIBUTED RANDOM NUMBERS
RCPA	READ (A PORTION OF) CONTROL POINT AREA
RECOVRD	ON RECOVERY, PRINT EXCHANGE JUMP PACKAGE, RA+0 THRU RA+77B
REDUCE	REDUCE FL TO MINIMUM -OR- REQUEST ADDITIONAL FL RELATIVE TO START OF BLANK COMMON
REPLAC	REPLACE ONE CHARACTER WITH ANOTHER IN AN ARRAY
REPLACM	REPLACE SEVERAL CHARACTERS WITH OTHER CHARACTERS
REPLHI	REPLACE ALL CHARACTERS GREATER THAN SPECIFIED CHARACTER WITH NEW CHARACTER
REPLLO	REPLACE ALL CHARACTERS LESS THAN SPECIFIED CHARACTER WITH NEW CHARACTER
REPLNE	REPLACE ALL CHARACTERS (EXCEPT SPECIFIED CHARACTER) WITH A SPECIFIED CHARACTER
RFFT	FAST FOURIER TRANSFORM FOR REAL TABULATED DATA
RFSN	REVERSE FAST FOURIER TRANSFORM
RNDMIZ	EMULATE BASIC LANGUAGE 'RANDOMIZE' STATEMENT (CAN BE USED TO GUARANTEE FIRST CALL TO RANF WILL RESULT IN A DIFFERENT NUMBER WITH EACH EXECUTION OF A PROGRAM)
RN1	UNIFORM RANDOM NUMBER USING TWO CONGRUENTIAL GENERATORS
RN2	UNIFORM RANDOM NUMBER USING ONE CONGRUENTIAL GENERATOR
ROOTER	GENERAL ROOT FINDER
ROUTERC	SUPPLY DESCRIPTION OF ROUTE RETURN CODE
RSO	RANK ORDER STANDARDIZED OBSERVATIONS
SBYT	STORE VARIABLE LENGTH BYTE

SEMICO	REPLACE DISPLAY CODE 00B WITH 77B (SEMI-COLON)
SETREW	CONVERT ALPHABETIC REWIND OPTION INTO RM OPEN AND CLOSE CODES
SHIFTA	SHIFT ARRAY A SPECIFIED NUMBER OF BITS (CROSSING OVER WORD BOUNDARIES)
SIMP	SIMPSON'S RULE INTEGRATION
SIMPUN	SIMPSON'S RULE INTEGRATION - UNEQUAL INTERVALS
SKPFIL	REPOSITION A SEQUENTIAL FILE FORWARD OR BACKWARD BY A SPECIFIED NUMBER OF UNITS (FOR EXISTING RECORDS ONLY)
SKPSTAT	GET THE STATUS OF THE LAST CALL TO 'SKPFIL'
SKWEZL	SQUEEZE LEFT AND REMOVE BLANKS AND 00B
SKWEZR	SQUEEZE RIGHT AND REMOVE BLANKS AND 00B
SMOOTH	LEAST SQUARES POLYNOMIAL SMOOTHING
SNCNDN	JACOBIAN ELLIPTIC FUNCTION
SOV	STANDARDIZATION OF VARIABLES - STATISTICS
SPLFIT	SPLINE CURVE FIT
SQFIT	POLYNOMIAL LEAST SQUARE FIT
SR1	INITIAL STEPWISE REGRESSION ANALYSIS BASED ON BMD02R
SR2	ONE STEP IN STEPWISE REGRESSION ANALYSIS
SR3	COMPUTE RESIDUALS FROM SR2 REGRESSION
SSORT	FTN SHELL SORT
SSORTF	FTN CALLABLE SHELL SORT FOR TWO-DIMENSIONAL ARRAYS
SSORTI	FTN CALLABLE SHELL SORT FOR TWO-DIMENSIONAL ARRAYS
SSORTL	FTN LOGICAL SHELL SORT
SSORT3	FTN-CALLABLE SHELL SORT FOR REAL ARRAYS WITH ASSOCIATED REAL ARRAY AND INTEGER ARRAY
STUTEE	STUDENT'S T DISTRIBUTION
SUMIT	SUM ELEMENTS OF REAL ARRAY
SWAP	SWAP TWO ARRAYS

TEKTRI	INITIALIZE COMMON BLOCK /TEKTRN/ WITH ASCII CONTROL CODES FOR THE TEKTRONIX GRAPHICS TERMINALS
TIMLEFT	DETERMINE CP (AND IO) TIME LEFT SINCE START OF BATCH JOB OR INTERCOM COMMAND
TOV	TRANSFORMATION OF VARIABLES BY IDENTITY, LOG BASE 10, SQUARE ROOT, SQUARE
TRAILBZ	CHANGE TRAILING BLANKS TO ZEROS (OOB)
UNHEX3	SPREAD 2 CHARACTERS INTO 3 HEX DIGITS
UNLOAD	UNLOAD A FORTRAN FILE
VALDAT	LOGICAL FUNCTION TO VALIDATE A DATE FORMAT
VALIDT	VALIDATE AN ARRAY TO SEE THAT EACH ELEMENT IS ONE OF A USER-SPECIFIED LIST
VARAH1	EIGENVALUES AND EIGENVECTORS OF A GENERAL REAL MATRIX
VARAH2	IMPROVED ESTIMATES AND BOUNDS FOR EIGENSYSTEM OF A GENERAL REAL MATRIX
VFILL	FILL AN ARRAY WITH USER-SPECIFIED WORD
VT100I	INITIALIZE COMMON BLOCK /VT100/ WITH ASCII CONTROL CODES FOR THE DEC VT100 CRT
WARNING	FTN-CALLABLE 'WARNING' CONTROL CARD
WEKDAY	DETERMINE THE DAY OF THE WEEK FOR ANY GREGORIAN DATE FROM OCTOBER 15, 1582 THRU FEBRUARY 28, 4000
XFIL	FILON'S METHOD FOR INTEGRALS WITH SIN AND COS
ZBLANK	CHANGE BLANKS TO OOB AND VICE VERSA
ZEROFL	ZERO FIELD LENGTH (SECURITY EOJ)
ZEROS	REPLACE BLANKS WITH (DISPLAY CODE) ZEROS, MULTIPLE FIELDS
ZSYSEQ	FORTAN CALLABLE SYSTEM CALL

*** DESCRIPTIVE TITLES (NSRDC5) ***

SUBPROGRAMS IN LIBRARY 'NSRDC5' ARE LISTED ALPHABETICALLY BELOW.

AC	GET ACCOUNT NUMBER FOR THIS JOB
ALTYM	OBTAIN CPA, CPB, CP, PP, IO AND WALL CLOCK TIMES SINCE START OF JOB (OR INTERCOM SESSION)
BANR	PRINT A BANNER (LETTERS ARE 10 LINES HIGH, LINES ARE 131 PRINT POSITIONS LONG)
BANR6	PRINT A BANNER (LETTERS ARE 6 LINES HIGH, LINES ARE 80 PRINT POSITIONS LONG)
CENTER	CENTER A CHARACTER STRING
CFIND	SCAN CHARACTER ARRAY FOR CHARACTER WORD
CHIN	CONVERT I-FORMATTED CHARACTER STRING TO INTEGER
CMMDUMP	DUMP COMMON MEMORY MANAGER (CMM) DYNAMIC AREA HEADERS AND
CMMMERC	SUPPLY DESCRIPTION OF CMM MEMORY ERROR CODE TRAILER WITH OPTIONAL DUMP OF THE CONTENTS OF EACH BLOCK
CMMPGFS	PRINT THE LARGEST BLOCK-SIZES AVAILABLE FOR ALL POSSIBLE CONDITIONS
CMMPGOS	PRINT THE CONTENTS OF THE ARRAY RETURNED BY SUBROUTINE CMMGOS
CMMPGSS	PRINT THE CONTENTS OF THE ARRAY RETURNED BY SUBROUTINE CMMGSS
CMMUERC	SUPPLY DESCRIPTION OF CMM USER ERROR CODE
CSHUFL	SHUFFLE A CHARACTER ARRAY
CSORT	SORT A CHARACTER ARRAY
CSORTD	SORT A CHARACTER ARRAY (DESCENDING)
CSORTN	SORT A CHARACTER ARRAY (HAVING AN ASSOCIATED NON-CHARACTER ARRAY)
CSORT2	SORT A CHARACTER ARRAY (HAVING AN ASSOCIATED CHARACTER ARRAY)
CVCHIN	CONVERT I-FORMATTED CHARACTER STRING TO INTEGER
CVCHOL	CONVERT CHARACTER STRING TO HOLLERITH STRING
CVHOCH	CONVERT HOLLERITH STRING TO CHARACTER STRING
CVINCH	CONVERT INTEGER TO CHARACTER STRING

DMPCPA	SHORT DUMP OF JOB CONTROL POINT AREA
DUMPXPK	DUMP EXCHANGE PACKAGE (REGISTERS, POINTERS, ETC.)
ELTYM	OBTAIN CPA, CPB, CP, PP, IO AND WALL CLOCK TIMES SINCE LAST CALL
FIRSTCH	FIND FIRST NON-BLANK IN CHARACTER VARIABLE
FRSTCH	FIND FIRST NON-BLANK IN CHARACTER VARIABLE
GETSTR	EXTRACT CHARACTER STRING ACCORDING TO USER-DEFINED CRITERIA
HMS2S	CONVERT HH.MM.SS TO SECONDS
IDID	GET USER INITIALS AND INTERCOM USER ID FROM CHARGE CARD OR LOGIN
ITRANS	TRANSLATE CHARACTERS ACCORDING TO USER-SPECIFIED TRANSLATE TABLES
JOBORG	DETERMINE JOB ORIGIN
LASTCH	DETERMINE NUMBER OF CHARACTERS THRU LAST NON-BLANK
LASTCHH	DETERMINE NUMBER OF CHARACTERS THRU LAST NON-BLANK IN A HOLLERITH WORD OR ARRAY
LEFT	LEFT-JUSTIFY A CHARACTER STRING
LSTCH	DETERMINE NUMBER OF CHARACTERS THRU LAST NON-BLANK
MFRAME	OBTAIN THE MACHINE AND MAINFRAME RUNNING THE PROGRAM
MF2CPU	RETURN CPU NAME CORRESPONDING TO SUPPLIED MAINFRAME NAME
NEWDAT	ADD/SUBTRACT SPECIFIED NUMBER OF DAYS TO/FROM A GIVEN DATE
NUMER	TEST STRING FOR NUMERICS
PFRG	SUPPLY DESCRIPTION OF PERMANENT FILE FUNCTION RETURN CODE
PM	WRITE 'PM' PRINTER MESSAGE
PRTYM	GET AND PRINT CPA, CPB, CP, PP, IO AND WALL CLOCK TIMES SINCE LAST CALL AND PRINT USER-SUPPLIED MESSAGE
RIGHT	RIGHT-JUSTIFY A CHARACTER STRING
ROUTERC	SUPPLY DESCRIPTION OF ROUTE RETURN CODE

*** DESCRIPTIVE TITLES (NSRDC5) ***

SUBPROGRAMS IN LIBRARY 'NSRDC5' ARE LISTED ALPHABETICALLY BELOW.

AC	GET ACCOUNT NUMBER FOR THIS JOB
ALTYM	OBTAIN CPA, CPB, CP, PP, IO AND WALL CLOCK TIMES SINCE START OF JOB (OR INTERCOM SESSION)
BANR	PRINT A BANNER (LETTERS ARE 10 LINES HIGH, LINES ARE 131 PRINT POSITIONS LONG)
BANR6	PRINT A BANNER (LETTERS ARE 6 LINES HIGH, LINES ARE 80 PRINT POSITIONS LONG)
CENTER	CENTER A CHARACTER STRING
CFIND	SCAN CHARACTER ARRAY FOR CHARACTER WORD
CHIN	CONVERT I-FORMATTED CHARACTER STRING TO INTEGER
CMMDUMP	DUMP COMMON MEMORY MANAGER (CMM) DYNAMIC AREA HEADERS AND
CMMMERC	SUPPLY DESCRIPTION OF CMM MEMORY ERROR CODE TRAILER WITH OPTIONAL DUMP OF THE CONTENTS OF EACH BLOCK
CMMPGFS	PRINT THE LARGEST BLOCK-SIZES AVAILABLE FOR ALL POSSIBLE CONDITIONS
CMMPGOS	PRINT THE CONTENTS OF THE ARRAY RETURNED BY SUBROUTINE CMMGOS
CMMPGSS	PRINT THE CONTENTS OF THE ARRAY RETURNED BY SUBROUTINE CMMGSS
CMMUERC	SUPPLY DESCRIPTION OF CMM USER ERROR CODE
CSHUFL	SHUFFLE A CHARACTER ARRAY
CSORT	SORT A CHARACTER ARRAY
CSORTD	SORT A CHARACTER ARRAY (DESCENDING)
CSORTN	SORT A CHARACTER ARRAY (HAVING AN ASSOCIATED NON-CHARACTER ARRAY)
CSORT2	SORT A CHARACTER ARRAY (HAVING AN ASSOCIATED CHARACTER ARRAY)
CVCHIN	CONVERT I-FORMATTED CHARACTER STRING TO INTEGER
CVCHOL	CONVERT CHARACTER STRING TO HOLLERITH STRING
CVHOCH	CONVERT HOLLERITH STRING TO CHARACTER STRING
CVINCH	CONVERT INTEGER TO CHARACTER STRING

DMPCPA SHORT DUMP OF JOB CONTROL POINT AREA

DUMPPXPK DUMP EXCHANGE PACKAGE (REGISTERS, POINTERS, ETC.)

ELTYM OBTAIN CPA, CPB, CP, PP, IO AND WALL CLOCK TIMES SINCE LAST
 CALL

FIRSTCH FIND FIRST NON-BLANK IN CHARACTER VARIABLE

FRSTCH FIND FIRST NON-BLANK IN CHARACTER VARIABLE

GETSTR EXTRACT CHARACTER STRING ACCORDING TO USER-DEFINED CRITERIA

HMS2S CONVERT HH.MM.SS TO SECONDS

IDID GET USER INITIALS AND INTERCOM USER ID FROM CHARGE CARD OR
 LOGIN

ITRANS TRANSLATE CHARACTERS ACCORDING TO USER-SPECIFIED TRANSLATE
 TABLES

JOBORG DETERMINE JOB ORIGIN

LASTCH DETERMINE NUMBER OF CHARACTERS THRU LAST NON-BLANK

LASTCHH DETERMINE NUMBER OF CHARACTERS THRU LAST NON-BLANK IN A
 HOLLERITH WORD OR ARRAY

LEFT LEFT-JUSTIFY A CHARACTER STRING

LSTCH DETERMINE NUMBER OF CHARACTERS THRU LAST NON-BLANK

MFRAME OBTAIN THE MACHINE AND MAINFRAME RUNNING THE PROGRAM

MF2CPU RETURN CPU NAME CORRESPONDING TO SUPPLIED MAINFRAME NAME

NEWDAT ADD/SUBTRACT SPECIFIED NUMBER OF DAYS TO/FROM A GIVEN DATE

NUMER TEST STRING FOR NUMERICS

PFR SUPPLY DESCRIPTION OF PERMANENT FILE FUNCTION RETURN CODE

PM WRITE 'PM' PRINTER MESSAGE

PRTYM GET AND PRINT CPA, CPB, CP, PP, IO AND WALL CLOCK TIMES SINCE
 LAST CALL AND PRINT USER-SUPPLIED MESSAGE

RIGHT RIGHT-JUSTIFY A CHARACTER STRING

ROUTERC SUPPLY DESCRIPTION OF ROUTE RETURN CODE

SETREW CONVERT REWIND OPTION INTO OPEN AND CLOSE CODES

SM5PRNT PRINT CONTENTS OF SORT/MERGE 5 STATISTICS ARRAY

S2HMS CONVERT SECONDS TO ' HH.MM.SS. '

TRANS TRANSLATE CHARACTERS ACCORDING TO USER-SPECIFIED TRANSLATE
TABLES

TTYMSG DRIVER TO WRITE A LINE TO AN INTERACTIVE TERMINAL

TTYOPN OPEN INTERACTIVE INPUT AND OUTPUT FILES

***** SUBPROGRAM DOCUMENTATION (NSRDC) *****

THIS CHAPTER CONTAINS THE MACHINE-READABLE DOCUMENTATION FOR MANY SUBPROGRAMS IN LIBRARY 'NSRDC'. NON-MACHINE-READABLE DOCUMENTATION FOR OTHER ROUTINES IN THE LIBRARY IS ON FILE IN USER SERVICES, CODE 1892.1, (202) 227-1907.

ALL DOCUMENT FILES RESIDE ON THE MASS STORAGE SYSTEM (MSS). YOUR MSACCES PASSWORD MUST BE SUBMITTED TO THE SYSTEM BEFORE DOCUMENTS CAN BE OBTAINED. THIS MAY BE DONE WITH A SEPARATE 'MSACCES' COMMAND OR BY USING THE MSACCES PARAMETER IN THE BEGIN STATEMENT.

*** HOW TO PRINT A DOCUMENT ***

INDIVIDUAL DOCUMENTS MAY BE PRINTED USING:

BEGIN,DOCGET,,NSRDC,<SUBPROG>,OUTPUT,MSACCES=<PASSWORD>.

WHERE <SUBPROG> IS THE DESIRED DOCUMENT.

SEVERAL DOCUMENTS MAY BE PRINTED AT ONE TIME USING:

BEGIN,DOCGET,,NSRDC,,,OUTPUT,,,DOCS,MSACCES=<PASSWORD>.

WHERE DOCS IS A FILE CONTAINING THE NAMES OF THE DESIRED DOCUMENTS:

<SUBPROG1>,<SUBPROG2>,...,<SUBPROGI>
<SUBPROGI+1>,...,<SUBPROGN>

ALL DOCUMENTS MAY BE PRINTED USING:

BEGIN,DOCGET,,NSRDC,,ALL,OUTPUT,MSACCES=<PASSWORD>.

IF YOU HAVE ALREADY SUBMITTED YOUR MSACCES PASSWORD IN THE JOB OR INTERACTIVE SESSION, IT MAY BE OMITTED FROM THE 'BEGIN,DOCGET,.....'.

TO PRINT THE DOCUMENT(S) ON THE XEROX 8700, EITHER:

A) ADD 'FID=<FID>' TO THE 'BEGIN,DOCGET,.....'
WHERE <FID> IS THE FILE ID FOR THE BANNER

B) USE
BEGIN,XEROX,,OUTPUT,FID,,DOCPRT.

SUBROUTINE 'AC'
FUNCTION 'AC'

PURPOSE

GET ACCOUNT NUMBER FOR THIS JOB

FUNCTIONAL CATEGORIES: Q0

LANGUAGE: FORTRAN IV

COMPUTERS (OPERATING SYSTEMS)
CDC 6000/CYBER 170 (NOS/BE)

REMARKS

'AC' MUST BE DECLARED INTEGER IN THE CALLING ROUTINE.

USAGE

CALL AC (I)
IVARIABLE = AC (I)

DESCRIPTION OF PARAMETERS

AC - WILL CONTAIN ACCOUNT NUMBER
(INTEGER TYPE VARIABLE)
I - WILL ALSO CONTAIN ACCOUNT NUMBER

CM REQUIRED: 36B

SUBPROGRAMS REQUIRED

PART OF LANGUAGE
NONE
OTHERS
RCPA - READ CONTROL POINT AREA

ARITHMETIC STATEMENT FUNCTIONS

L71FMT - FAST L-FORMAT DECODE (LEFT-ADJ, ZERO-FILLED)

METHOD

THE ACCOUNT NUMBER IS TAKEN FROM CONTROL POINT AREA.

AUTHOR

DAVID V SOMMER - DTNSRDC CODE 1892.2

DATE WRITTEN: 12/04/75

DATE(S) REVISED

02/27/76
03/23/83 - RECOMPILE AT NOS/BE LEVEL 552

LOCATION OF DECKS

SOURCE
UPDATE LIBRARY ON MSS: NSRDCPL,UN=CSYS
OBJECT
EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC,,AC,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'ADJL'

PURPOSE

LEFT ADJUST A LINE OF WORDS LEAVING ONE SPACE BETWEEN WORDS

FUNCTIONAL CATEGORIES: M4

LANGUAGE: FORTRAN IV

COMPUTERS (OPERATING SYSTEMS)
CDC 6000/CYBER 170 (NOS/BE)

REMARKS
NONE

USAGE
CALL ADJL (A, NA, NC, NW, NWORDS)

DESCRIPTION OF PARAMETERS

A - ARRAY CONTAINING WORDS TO BE LEFT-ADJUSTED
(WILL BE REPLACED BY LEFT-ADJUSTED ARRAY)
NA - NUMBER OF COMPUTER WORDS IN 'A' (DIMENSION OF 'A')
NC - OUTPUT NUMBER OF CHARACTERS
NW - OUTPUT NUMBER OF COMPUTER WORDS
(SUBSCRIPT OF LAST NON-BLANK WORD IN 'A')
NWORDS - OUTPUT NUMBER OF WORDS IN LINE

CM REQUIRED: 126B

SUBPROGRAMS REQUIRED

PART OF LANGUAGE
NONE

OTHERS

GETCHA - GET CHARACTER FROM ARRAY
PUTCHA - PUT CHARACTER INTO ARRAY

AUTHOR

DAVID V SOMMER - DTNSRDC CODE 1892.2

DATE WRITTEN: 03/24/76

DATE(S) REVISED

03/23/83 - RECOMPILE AT NOS/BE LEVEL 552

LOCATION OF DECKS

SOURCE

UPDATE LIBRARY ON MSS: NSRDCPL,UN=CSYS

OBJECT

EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC,,ADJL,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'ADJR'

PURPOSE

RIGHT ADJUST A LINE OF WORDS LEAVING ONE SPACE BETWEEN WORDS

FUNCTIONAL CATEGORIES: M4

LANGUAGE: FORTRAN IV

COMPUTERS (OPERATING SYSTEMS)

CDC 6000/CYBER 170 (NOS/BE)

REMARKS

NONE

USAGE

CALL ADJR (A, NA, NC, NW, NWORDS)

DESCRIPTION OF PARAMETERS

A - ARRAY CONTAINING WORDS TO BE RIGHT-ADJUSTED
(WILL BE REPLACED BY RIGHT-ADJUSTED ARRAY)
NA - NUMBER OF COMPUTER WORDS IN 'A' (DIMENSION OF 'A')
NC - OUTPUT POSITION OF FIRST NON-BLANK CHARACTER
NW - OUTPUT SUBSCRIPT OF FIRST NON-BLANK WORD IN 'A'
NWORDS - OUTPUT NUMBER OF WORDS IN LINE

CM REQUIRED: 137B

SUBPROGRAMS REQUIRED

PART OF LANGUAGE

NONE

OTHERS

GETCHA - GET CHARACTER FROM ARRAY
PUTCHA - PUT CHARACTER INTO ARRAY

AUTHOR

DAVID V SOMMER - DTNSRDC CODE 1892.2

DATE WRITTEN: 03/24/76

DATE(S) REVISED

03/23/83 - RECOMPILE AT NOS/BE LEVEL 552

LOCATION OF DECKS

SOURCE

UPDATE LIBRARY ON MSS: NSRDCPL,UN=CSYS

OBJECT

EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC,,ADJR,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'ALTIME'

PURPOSE

OBTAIN CPA, CPB, CP, PP, IO AND WALL CLOCK TIMES SINCE
START OF JOB (OR INTERCOM SESSION)

FUNCTIONAL CATEGORIES: QO NO

LANGUAGE: FORTRAN IV

COMPUTERS (OPERATING SYSTEMS)
CDC 6000/CYBER 170 (NOS/BE)

REMARKS
NONE

USAGE
CALL ALTIME (TIMES)

DESCRIPTION OF PARAMETER

TIMES - 7-WORD ARRAY TO CONTAIN THE FOLLOWING:

- 1 - CPA TIME IN SECONDS
- 2 - CPB TIME IN SECONDS
- 3 - CP TIME IN SECONDS (CPA+CPB)
- 4 - PP TIME IN SECONDS
- 5 - IO TIME IN SECONDS
- 6 - WALL CLOCK TIME (HH.MM.SS.)
- 7 - WALL CLOCK TIME IN SECONDS

CM REQUIRED: 61B

SUBPROGRAMS REQUIRED

PART OF LANGUAGE

NONE

OTHERS

ISEC - CONVERT HH.MM.SS TO SECONDS

RCPA - READ CONTROL POINT AREA

ARITHMETIC STATEMENT FUNCTIONS

R65FMT - FAST R-FORMAT DECODE (RIGHT-ADJ, ZERO-FILLED)

AUTHOR

DAVID V SOMMER - DTNSRDC CODE 1892.2

DATE WRITTEN: 12/15/75

DATE(S) REVISED

03/23/83 - RECOMPILE AT NOS/BE LEVEL 552

LOCATION OF DECKS

SOURCE

UPDATE LIBRARY ON MSS: NSRDCPL,UN=CSYS

OBJECT

EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOUGET,,NSRDC,,ALTIME,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'ASCADD'

PURPOSE

ADD AN ASCII STRING TO ANOTHER ASCII STRING

FUNCTIONAL CATEGORIES: M4

LANGUAGE: FORTRAN 77

COMPUTERS (OPERATING SYSTEMS)

CDC 6000/CYBER 170 (NOS/BE)

REMARKS

SUBROUTINE 'ASCII' INSERTS ANY NUMBER OF ASCII STRINGS. IT DOES THIS BY CALLING 'ASCADD' TO ADD EACH STRING. 'ASCADD' MAY BE CALLED DIRECTLY TO ADD A SINGLE STRING.

SEE ALSO SUBROUTINES ASCADM, ASCBSX, ASCGET, ASCII, ASCII1, ASCLEN, ASCPUT, OMRONI, TEKTRI, VT1001, AND PROCEDURE ASCII0.

USAGE

CALL ASCADD (BUF, NEXT, ADD)

DESCRIPTION OF PARAMETERS

BUF - INTEGER BUFFER ARRAY. EACH WORD HOLDS FIVE ASCII CHARACTERS.

NEXT - CHARACTER POSITION IN THE ARRAY TO START ADDING THE NEW CHARACTERS (1 IS THE LEFT-MOST POSITION IN THE ARRAY). THIS MUST BE AN INTEGER VARIABLE BECAUSE IT WILL BE INCREMENTED BY THE NUMBER OF CHARACTERS ADDED BY THE CALL, THUS POINTING TO THE NEXT CHARACTER POSITION AVAILABLE FOR THE NEXT CALL. ON RETURN, THE CHARACTER AT THIS POSITION IS SET TO 12 BITS OF BINARY ZERO.

ADD - THE ASCII CHARACTER STRING TO BE ADDED. THE STRING STARTS IN CHARACTER POSITION 1 AND CONTINUES UNTIL A 12-BIT BINARY ZERO IS ENCOUNTERED.

CM REQUIRED: 33B

SUBPROGRAMS REQUIRED

PART OF LANGUAGE

NONE

OTHERS

ASCGET - GET AN ASCII CHARACTER

ASCPUT - PUT AN ASCII CHARACTER

AUTHOR

DAVID V. SOMMER - DTNSRDC CODE 1892.2

DATE WRITTEN: 03/13/84

DATE(S) REVISED

LOCATION OF DECKS

SOURCE DECK

UPDATE LIBRARY ON MSS: NSRDCPL,UN=CSYS

OBJECT DECK

EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC,,ASCADD,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'ASCADM'

PURPOSE

ADD AN ASCII STRING TO ANOTHER ASCII STRING MULTIPLE TIMES

FUNCTIONAL CATEGORIES: M4

LANGUAGE: FORTRAN 77

COMPUTERS (OPERATING SYSTEMS)

CDC 6000/CYBER 170 (NOS/BE)

REMARKS

SEE ALSO SUBROUTINES ASCADD, ASCBSX, ASCGET, ASCII, ASCIIII, ASCLN, ASCPUT, OMRONI, TEKTRI, VT100I, AND PROCEDURE ASCIIIO.

USAGE

CALL ASCADM (BUF, NEXT, ADD, NTIMES)

DESCRIPTION OF PARAMETERS

BUF - INTEGER BUFFER ARRAY. EACH WORD HOLDS FIVE ASCII CHARACTERS.

NEXT - CHARACTER POSITION IN THE ARRAY TO START ADDING THE NEW CHARACTERS (1 IS THE LEFT-MOST POSITION IN THE ARRAY). THIS MUST BE AN INTEGER VARIABLE BECAUSE IT WILL BE INCREMENTED BY THE NUMBER OF CHARACTERS ADDED BY THE CALL, THUS POINTING TO THE NEXT CHARACTER POSITION AVAILABLE FOR THE NEXT CALL. ON RETURN, THE CHARACTER AT THIS POSITION IS SET TO 12 BITS OF BINARY ZERO.

ADD - THE ASCII CHARACTER STRING TO BE ADDED. THE STRING STARTS IN CHARACTER POSITION 1 AND CONTINUES UNTIL A 12-BIT BINARY ZERO IS ENCOUNTERED.

NTIMES - NUMBER OF TIMES STRING 'ADD' IS TO BE ADDED.

CM REQUIRED: 43B

SUBPROGRAMS REQUIRED

PART OF LANGUAGE

NONE

OTHERS

ASCADD - ADD AN ASCII STRING

AUTHOR

DAVID V. SOMMER - DTNSRDC CODE 1892.2

DATE WRITTEN: 03/19/84

DATE(S) REVISED

LOCATION OF DECKS

SOURCE DECK

UPDATE LIBRARY ON MSS: NSRDCPL,UN=CSYS

OBJECT DECK

EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC,,ASCADM,OUTPUT,MSACCES=<PASSWORD>.

INTEGER FUNCTION 'ASCBSX'

PURPOSE

REMOVE BS (BACKSPACE) AND CAN (CTRL-X) FROM A STRING

FUNCTIONAL CATEGORIES: M4

LANGUAGE: FORTRAN 77 EXTENDED

COMPUTERS (OPERATING SYSTEMS)

CDC 6000/CYBER 170 (NOS/BE)

REMARKS

THE STRING ENDS WITH A CARRIAGE RETURN (CR) OR BINARY ZERO.
CR, IF PRESENT, IS CHANGED TO BINARY ZERO.

SEE ALSO SUBROUTINES ASCADD, ASCADM, ASCGET, ASCII, ASCIII,
ASCLN, ASCPUT, OMRONI, TEKTRI, VT100I, AND PROCEDURE
ASCII0.

USAGE

INTEGER ASCBSX

...

ASCBSX (S1, S2)

DESCRIPTION OF PARAMETERS

S1 - INPUT ASCII STRING TO BE PROCESSED

S2 - OUTPUT ASCII STRING WITH BS/CAN REMOVED

(S1 AND S2 MAY BE THE SAME)

ASCBSX - WILL HAVE THE LENGTH OF THE OUTPUT STRING

(I.E., THE POSITION OF THE NEXT CHARACTER AFTER
THE STRING)

CM REQUIRED:

SUBPROGRAMS REQUIRED

PART OF LANGUAGE

NONE

OTHERS

ASCGET - GET AN ASCII CHARACTER

ASCPUT - PUT AN ASCII CHARACTER

AUTHOR

DAVID V. SOMMER - DTNSRDC CODE 1892.2

DATE WRITTEN: 03/21/84

DATE(S) REVISED

LOCATION OF DECKS

SOURCE DECK

UPDATE LIBRARY ON MSS: NSRDCPL,UN=CSYS

OBJECT DECK

EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC,,ASCBSX,OUTPUT,MSACCES=<PASSWORD>.

INTEGER FUNCTION 'ASCGET'

PURPOSE

GET AN ASCII CHARACTER FROM AN ASCII STRING

FUNCTIONAL CATEGORIES: M4

LANGUAGE: FORTRAN 77 EXTENDED

COMPUTERS (OPERATING SYSTEMS)
CDC 6000/CYBER 170 (NOS/BE)

REMARKS

SEE ALSO SUBROUTINES ASCADD, ASCADM, ASCBSX, ASCII, ASCIIII,
ASCLN, ASCPUT, OMRONI, TEKTRI, VT100I, AND PROCEDURE
ASCIIIO.

USAGE

INTEGER ASCGET

...
ASCGET (FIELD, N)

DESCRIPTION OF PARAMETERS

FIELD - ASCII STRING FROM WHICH CHARACTER IS TO BE
EXTRACTED.

N - CHARACTER POSITION IN FIELD TO BE EXTRACTED. (1 IS
THE LEFT-MOST POSITION IN THE STRING). THIS MUST
BE AN INTEGER VARIABLE BECAUSE IT WILL BE
INCREMENTED BY ONE, THUS POINTING TO THE NEXT
CHARACTER POSITION AVAILABLE FOR THE NEXT CALL.

CM REQUIRED: 30B

SUBPROGRAMS REQUIRED

PART OF LANGUAGE
AND MOD OR SHIFT
OTHERS
NONE

AUTHOR

DAVID V. SOMMER - DTNSRDC CODE 1892.2

DATE WRITTEN: 03/13/84

DATE(S) REVISED

LOCATION OF DECKS

SOURCE DECK
UPDATE LIBRARY ON MSS: NSRDCPL,UN=CSYS
OBJECT DECK
EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC,,ASCGET,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'ASCII'

PURPOSE

CREATE AN ASCII MESSAGE FROM STRINGS OF ASCII CHARACTERS

FUNCTIONAL CATEGORIES: M4

LANGUAGE: CDC CP COMPASS

COMPUTERS (OPERATING SYSTEMS)

CDC 6000/CYBER 170 (NOS/BE)

REMARKS

THIS SUBROUTINE IS USED TO CREATE ASCII MESSAGES TO BE DISPLAYED AT AN INTERACTIVE TERMINAL. THIS ALLOWS THE GENERATION OF MESSAGES IN UPPER AND LOWER CASE AND THE TAILORING OF OUTPUT TO CERTAIN "INTELLIGENT" TERMINALS, SUCH AS DEC VT100, OMRON, ETC.

BEFORE PREPARING A PROGRAM TO USE THIS FEATURE, YOU SHOULD RUN PROCEDURE 'ASCII0' TO GET THE REQUIRED LABELLED COMMON BLOCK(S):

```
/ASCII/      <-- ASCII CHARACTERS
/OMRON/      <-- OMRON CONTROL CODES
/TEKTRN/     <-- TEKTRONIX CONTROL CODES
/VT100/      <-- VT100 CONTROL CODES
```

THESE MUST BE MERGED INTO ANY (SUB)PROGRAM WHICH WILL GENERATE ASCII MESSAGES.

THE PROGRAM SHOULD START BY INITIALIZING THE LABELLED COMMON BLOCKS USED:

```
CALL ASCII      <-- INITIALIZE /ASCII/
CALL OMRONI     <-- INITIALIZE /OMRON/
CALL TEKTRI     <-- INITIALIZE /TEKTRN/
CALL VT100I     <-- INITIALIZE /VT100/
```

TO CREATE AN ASCII MESSAGE, ALLOCATE AN INTEGER BUFFER ARRAY CLEARING IT TO BINARY ZERO. THEN CALL ASCII ANY NUMBER OF TIMES TO PUT THE ASCII CHARACTERS INTO THE BUFFER. WHEN COMPLETE, WRITE THE BUFFER UNFORMATTED TO A SPECIALLY CONNECTED FILE.

TO CREATE ANOTHER ASCII MESSAGE, CLEAR THE BUFFER ARRAY TO BINARY ZERO, OR USE ANOTHER BUFFER ARRAY (ALSO CLEARED TO BINARY ZERO).

SEE ALSO SUBROUTINES ASCADD, ASCADM, ASCBSX, ASCGET, ASCII1, ASCLEN, ASCPUT, OMRONI, TEKTRI, VT100I, AND PROCEDURE ASCII0.

USAGE

CALL ASCII (BUF, NEXT, ASCSTR1, ASCSTR2, ...)

DESCRIPTIC OF PARAMETERS

- BUF - INTEGER BUFFER ARRAY. EACH WORD WILL HOLD FIVE ASCII CHARACTERS.
- NEXT - CHARACTER POSITION IN THE ARRAY TO START ADDING THE NEW CHARACTERS (1 IS THE LEFT-MOST POSITION IN THE ARRAY). THIS MUST BE AN INTEGER VARIABLE BECAUSE IT WILL BE INCREMENTED BY THE NUMBER OF CHARACTERS ADDED BY THE CALL, THUS POINTING TO THE NEXT CHARACTER POSITION AVAILABLE FOR THE NEXT CALL. ON RETURN, THE CHARACTER AT THIS POSITION IS SET TO 12 BITS OF BINARY ZERO.
- ASCSTR1 - FIRST ASCII CHARACTER STRING TO BE ADDED. THE STRING STARTS IN CHARACTER POSITION 1 AND CONTINUES UNTIL A 12-BIT BINARY ZERO IS ENCOUNTERED.
- ASCSTR2 - SECOND ASCII CHARACTER STRING TO BE ADDED.
- ... - ANY NUMBER OF ADDITIONAL ASCII CHARACTER STRINGS TO BE ADDED. THE ONLY LIMIT IS THE NUMBER OF FORTRAN CONTINUATION STATEMENTS ALLOWED.

CM REQUIRED: 21B

EXAMPLE

CREATE AND TYPE THE ASCII MESSAGE "GOOD MORNING!" ('G' AND 'M' ARE UPPER CASE; THE REST LOWER CASE). START BY CREATING "GOOD " AND "MORNING" INTO SEPARATE AREAS:

```
...
INTEGER BUF(128)      <-- WILL HOLD 640 ASCII CHARACTERS
INTEGER GOOD(2)       <-- WILL HOLD "GOOD "
INTEGER MORN(2)       <-- WILL HOLD "MORNING"
COMMON /ASCII/ ...    <-- FROM ASCII0 (SEE REMARKS)
DATA BUF/ 128 * 0/    <-- CLEAR TO BINARY ZERO
...
CALL ASCII            <-- INITIALIZE /ASCII/
...
COL = 1              <-- SET POINTER FOR FIRST CALL
CALL ASCII (GOOD, COL, GU, OL, OL, DL, SPACE)
COL = 1
CALL ASCII (MORN, COL, MU, OL, RL, NL, IL, NL, GL)
COL = 1
CALL ASCII (BUF, COL, CRLF, GOOD, MORN, EXCLAM, CRLF)
OPEN (5, FILE='YYYYTTY')
CALL CONNEC (5, 2)    <-- SET FOR 256-CHARACTER ASCII
WRITE (5) BUF         <-- WRITE OUT THE MESSAGE
...
CLOSE (5, STATUS='DELETE') <-- RETURN THE CONNECTED
                           LOCAL FILE
...
```

SUBPROGRAMS REQUIRED

PART OF LANGUAGE

NONE

OTHERS

ASCADD - ADD ASCII STRING

AUTHOR

STANLEY WILLNER - DTNSRDC CODE 1892.1

DATE WRITTEN: 03/09/84

DATE(S) REVISED

LOCATION OF DECKS

SOURCE DECK

UPDATE LIBRARY ON MSS: NSRDCPL,UN=CSYS

OBJECT DECK

EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC,,ASCII,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'ASCIII'

PURPOSE

INITIALIZE COMMON BLOCK /ASCII/ WITH ASCII CHARACTERS

FUNCTIONAL CATEGORIES: M4

LANGUAGE: FORTRAN 77 EXTENDED

COMPUTERS (OPERATING SYSTEMS)

CDC 6000/CYBER 170 (NOS/BE)

REMARKS

THIS SUBROUTINE MUST BE EXECUTED PRIOR TO GENERATING ASCII MESSAGES USING SUBROUTINE ASCII.

COMMON BLOCK /ASCII/ IS OBTAINED BY RUNNING PROCEDURE ASCII0 AND INSERTING IT INTO EACH (SUB)PROGRAM WHICH WILL GENERATE ASCII MESSAGES.

IN ADDITION TO THE 128 ASCII CHARACTERS, THE CHARACTER-PAIR CARRIAGE RETURN/LINE FEED IS ALSO DEFINED.

USAGE

CALL ASCIII

CM REQUIRED: 6B

NAMES OF ASCII CHARACTERS

THE STANDARD ASCII NAMES ARE USED FOR THE NON-PRINTING CHARACTERS:

NUL, SOH, STX, ETX, EOT, ENQ, ACK, BEL, BS, HT, LF, VT, FF, CR, SO, SI, DLE, DC1, DC2, DC3, DC4, NAK, SYN, ETB, CAN, EM, SUB, ESC, FS, GS, RS, US, DEL

THE PRINTING CHARACTERS HAVE THE FOLLOWING NAMES:

AU, BU, CU, DU, ..., ZU (UPPER CASE LETTERS)
AL, BL, CL, DL, ..., ZL (LOWER CASE LETTERS)
ZERO (0)
ONE (1)
TWO (2)
THREE (3)
FOUR (4)
FIVE (5)
SIX (6)
SEVEN (7)
EIGHT (8)
NINE (9)

SPACE (SPACE OR BLANK) (SYNONYM: BLANK)
EXCLAM (EXCLAMATION MARK)
QUEST (QUESTION MARK)
QUOTE (QUOTATION MARK -- '"')
APOST (APOSTROPHE OR SINGLE QUOTE -- ''')
GRAVE (GRAVE ACCENT OR BACK APOSTROPHE)
POUND (POUND SIGN) (SYNONYMS: SHARP, NUMBER)

DOLLAR	(DOLLAR SIGN)
PCT	(PER CENT SIGN)
AMPER	(AMPERSAND)
LPAREN	(LEFT PARENTHESIS)
RPAREN	(RIGHT PARENTHESIS)
LBRACK	(LEFT SQUARE BRACKET)
RBRACK	(RIGHT SQUARE BRACKET)
LBRACE	(LEFT BRACE OR LEFT CURLY BRACKET)
RBRACE	(RIGHT BRACE OF RIGHT CURLY BRACKET)
PLUS	(PLUS SIGN)
MINUS	(MINUS SIGN) (SYNONYM: DASH)
LESS	(LESS THAN SIGN)
EQUAL	(EQUAL SIGN)
GTR	(GREATER THAN SIGN)
ASTER	(ASTERISK) (SYNONYM: STAR)
COMMA	(COMMA)
PERIOD	(PERIOD)
SLASH	(SLASH)
BSLASH	(BACKSLASH)
COLON	(COLON)
SEMI	(SEMI-COLON)
AT	(AT SIGN -- @)
CARAT	(CARAT -- ^)
UNDER	(UNDERSCORE OR UNDERLINE -- _)
TILDE	(TILDE)

IN ADDITION, THE FOLLOWING CHARACTER-PAIR IS DEFINED:
 CRLF (CARRIAGE RETURN/LINE FEED)

EXAMPLE

CREATE AND TYPE THE ASCII MESSAGE "GOOD MORNING!" ('G' AND 'M' ARE UPPER CASE; THE REST IN LOWER CASE). START BY CREATING "GOOD " AND "MORNING" IN SEPARATE AREAS:

```

...
INTEGER BUF(128)      <-- WILL HOLD 640 ASCII CHARACTERS
INTEGER GOOD(2)       <-- WILL HOLD "GOOD "
INTEGER MORN(2)       <-- WILL HOLD "MORNING"
COMMON /ASCII/ ...    <-- FROM MSS (SEE REMARKS)
DATA BUF/ 128 * 0/    <-- CLEAR TO BINARY ZERO
...
CALL ASCII            <-- INITIALIZE /ASCII/
...
COL = 1               <-- SET POINTER FOR FIRST CALL
CALL ASCII (GOOD, COL, GU, OL, OL, DL, SPACE)
COL = 1
CALL ASCII (MORN, COL, MU, OL, RL, NL, IL, NL, GL)
COL = 1
CALL ASCII (BUF, COL, CRLF, GOOD, MORN, EXCLAM, CRLF)
OPEN (5, FILE='YYYYTTY')
CALL CONNec (5, 2)    <-- SET FOR 256-CHARACTER ASCII
WRITE (5) BUF         <-- WRITE OUT THE MESSAGE
...
CLOSE (5, STATUS='DELETE') <-- RETURN THE CONNECTED
                           LOCAL FILE
...

```

SUBPROGRAMS REQUIRED
PART OF LANGUAGE
NONE
OTHERS
NONE

AUTHOR
DAVID V SOMMER - DTNSRDC CODE 1892.2

DATE WRITTEN: 03/07/84

DATE(S) REVISED

LOCATION OF DECKS
SOURCE DECK
UPDATE LIBRARY ON MSS: NSRDCPL,UN=CSYS
OBJECT DECK
EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT
BEGIN,DOCGET,,NSRDC,,ASCII11,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'ASCLEN'

PURPOSE

FIND LENGTH OF AN ASCII STRING

FUNCTIONAL CATEGORIES: M4

LANGUAGE: FORTRAN 77 EXTENDED

COMPUTERS (OPERATING SYSTEMS)

CDC 6000/CYBER 170 (NOS/BE)

REMARKS

THE STRING ENDS WITH A CARRIAGE RETURN (CR) OR BINARY ZERO.
CR, IF PRESENT, IS CHANGED TO BINARY ZERO.

IF ASCLEN PROCESSES A STRING WHICH HAS BEEN READ (RATHER
THAN CREATED), BE SURE THE INPUT STATEMENT LIST INCLUDES
ROOM FOR THE CARRIAGE RETURN. IF THIS IS NOT DONE, THE
COMPUTED LENGTH MAY BE INCORRECT.

SEE ALSO SUBROUTINES ASCADD, ASCADM, ASCBSX, ASCGET, ASCII,
ASCIIII, ASCPUT, OMRONI, TEKTRI, VT100I, AND PROCEDURE
ASCIIIO.

USAGE

INTEGER ASCLEN

...
ASCLEN (BUF)

DESCRIPTION OF PARAMETER

BUF - ASCII STRING WHOSE LENGTH IS TO BE FOUND

CM REQUIRED: 20B

SUBPROGRAMS REQUIRED

PART OF LANGUAGE

NONE

OTHERS

NONE

AUTHOR

DAVID V. SOMMER - DTNSRDC CODE 1892.2

DATE WRITTEN: 03/20/84

DATE(S) REVISED

LOCATION OF DECKS

SOURCE DECK

UPDATE LIBRARY ON MSS: NSRDCPL,UN=CSYS

OBJECT DECK

EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC,,ASCLEN,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'ASCPUT'

PURPOSE

ADD AN ASCII CHARACTER TO AN ASCII STRING

FUNCTIONAL CATEGORIES: M4

LANGUAGE: FORTRAN 77 EXTENDED

COMPUTERS (OPERATING SYSTEMS)

CDC 6000/CYBER 170 (NOS/BE)

REMARKS

SEE ALSO SUBROUTINES ASCADD, ASCADM, ASCBSX, ASCGET, ASCII, ASCII1, ASCPUT, OMRON1, TEKTR1, VT1001, AND PROCEDURE ASCII0.

USAGE

CALL ASCPUT (BUF, NEXT, ASCHAR)

DESCRIPTION OF PARAMETERS

BUF - INTEGER BUFFER ARRAY. EACH WORD HOLDS FIVE ASCII CHARACTERS.

NEXT - CHARACTER POSITION IN THE ARRAY TO ADD THE NEW CHARACTER (1 IS THE LEFT-MOST POSITION IN THE ARRAY). THIS MUST BE AN INTEGER VARIABLE BECAUSE IT WILL BE INCREMENTED BY ONE FOR THE CHARACTER ADDED BY THE CALL, THUS POINTING TO THE NEXT CHARACTER POSITION AVAILABLE FOR THE NEXT CALL.

ASCHAR - THE ASCII CHARACTER TO BE ADDED.
(LEFT-JUSTIFIED, ZERO-FILLED)

CM REQUIRED: 35B

SUBPROGRAMS REQUIRED

PART OF LANGUAGE
AND MOD OR SHIFT
OTHERS
NONE

AUTHOR

DAVID V. SOMMER - DTNSRDC CODE 1892.2

DATE WRITTEN: 03/13/84

DATE(S) REVISED

LOCATION OF DECKS

SOURCE DECK
UPDATE LIBRARY ON MSS: NSRDCPL,UN=CSYS
OBJECT DECK
EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC,,ASCPUT,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'ASCTXT'

PURPOSE

CONVERT A DISPLAY CODE STRING TO AN ASCII STRING AND PUT IT INTO AN ASCII BUFFER

FUNCTIONAL CATEGORIES: M4

LANGUAGE: FORTRAN 77 EXTENDED

COMPUTERS (OPERATING SYSTEMS)
CDC 6000/CYBER 170 (NOS/BE)

REMARKS

THIS SUBROUTINE WILL SAVE TIME CREATING ASCII MESSAGES.

SEE ALSO SUBROUTINES ASCADD, ASCADM, ASCBSX, ASCGET, ASCII, ASCII, ASCLN, ASCPUT, OMRONI, TEKTRI, VT100I, AND PROCEDURE ASCIIO.

USAGE

CALL ASCTXT (BUF, COL, DC, TYPE)

DESCRIPTION OF PARAMETERS

- BUF - INTEGER BUFFER ARRAY. EACH WORD WILL HOLD FIVE ASCII CHARACTERS.
- NEXT - CHARACTER POSITION IN THE ARRAY TO START ADDING THE NEW CHARACTERS (1 IS THE LEFT-MOST POSITION IN THE ARRAY). THIS MUST BE AN INTEGER VARIABLE BECAUSE IT WILL BE INCREMENTED BY THE NUMBER OF CHARACTERS ADDED BY THE CALL, THUS POINTING TO THE NEXT CHARACTER POSITION AVAILABLE FOR THE NEXT CALL. ON RETURN, THE CHARACTER AT THIS POSITION IS SET TO 12 BITS OF BINARY ZERO.
- DC - DISPLAY CODE CHARACTER STRING
- TYPE - TYPE OF CONVERSION DESIRED
- 1 - CONVERT TO LOWER CASE
 - 2 - CONVERT TO UPPER CASE
 - 3 - CONVERT FIRST CHARACTER TO UPPER CASE AND THE REST TO LOWER CASE
- IN ALL CASES, NON-ALPHABETICS ARE CONVERTED TO THEIR ASCII EQUIVALENT.

CM REQUIRED: 233B

EXAMPLE

CREATE AND TYPE THE ASCII MESSAGE "GOOD MORNING!" ('G' AND 'M' ARE UPPER CASE; THE REST LOWER CASE).

```

...
INTEGER BUF(128)      <-- WILL HOLD 640 ASCII CHARACTERS
COMMON /ASCII/ ...    <-- FROM ASCII0 (SEE REMARKS)
DATA BUF/ 128 * 0/    <-- CLEAR TO BINARY ZERO
...
CALL ASCII            <-- INITIALIZE /ASCII/
...
COL = 1               <-- SET POINTER FOR FIRST CALL
CALL ASCII (BUF, COL, CRLF)
CALL ASCTXT (BUF, COL, 'GOOD ', 3)
CALL ASCTXT (BUF, COL, 'MORNING', 3)
CALL ASCII (BUF, COL, EXCLAM, CRLF)
OPEN (5, FILE='YYYYTTY')
CALL CONNEX (5, 2)     <-- SET FOR 256-CHARACTER ASCII
WRITE (5) BUF          <-- WRITE OUT THE MESSAGE
...
CLOSE (5, STATUS='DELETE') <-- RETURN THE CONNECTED
                           LOCAL FILE
...

```

SUBPROGRAMS REQUIRED
 PART OF LANGUAGE
 ICCHAR LEN
 OTHERS
 ASCII - CREATE ASCII STRING

AUTHOR
 STANLEY WILLNER - DTNSRDC CODE 1892.1

DATE WRITTEN: 05/17/84

DATE(S) REVISED

LOCATION OF DECKS
 SOURCE DECK
 UPDATE LIBRARY ON MSS: NSRDCPL,UN=CSYS
 OBJECT DECK
 EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT
 BEGIN,DOCGET,,NSRDC,,ASCTXT,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'ASHIFT'

PURPOSE

SHIFT EACH INDIVIDUAL WORD OF AN ARRAY

FUNCTIONAL CATEGORIES: M4

LANGUAGE: FORTRAN IV EXTENDED

COMPUTERS (OPERATING SYSTEMS)

CDC 6000/CYBER 170 (NOS/BE)

REMARKS

SEE 'SHIFTA' TO SHIFT AN ENTIRE ARRAY.

USAGE

CALL ASHIFT (A, NA, NABITS)

DESCRIPTION OF PARAMETERS

A - ARRAY, EACH WORD OF WHICH IS TO BE SHIFTED
NA - NUMBER OF WORDS IN 'A' TO BE SHIFTED
NABITS - NUMBER OF BITS TO SHIFT EACH WORD
POSITIVE -- SHIFT LEFT CIRCULAR
NEGATIVE -- SHIFT RIGHT WITH SIGN PROPAGATION

CM REQUIRED: 14B

SUBPROGRAMS REQUIRED

PART OF LANGUAGE

SHIFT

OTHERS

NONE

AUTHOR

DAVID V SOMMER - DTNSRDC CODE 1892.2

DATE WRITTEN: 1973

DATE(S) REVISED

03/23/83 - RECOMPILE AT NOS/BE LEVEL 552

LOCATION OF DECKS

SOURCE

UPDATE LIBRARY ON MSS: NSRDCPL,UN=CSYS

OBJECT

EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC,,ASHIFT,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'ASORT'

PURPOSE

FTN ALPHANUMERIC SORT

FUNCTIONAL CATEGORIES: M1

LANGUAGE: FORTRAN IV

COMPUTERS (OPERATING SYSTEMS)

CDC 6000/CYBER 170 (NOS/BE)

REMARKS

NONE

USAGE

CALL ASORT (A, I, L, TEM, PT, COL, KEY, TRANA, KEYM)
CALL ASORT (A, I, L, TEM, PT, COL, KEY, 0, KEYM)
CALL ASORT (A, I, L, TEM, PT, COL, KEY, TRANA)
CALL ASORT (A, I, L, TEM, PT, COL, KEY)
CALL ASORT (A, I, L, TEM, PT, COL)

DESCRIPTION OF PARAMETERS

A - TWO-DIMENSIONAL ARRAY TO BE SORTED
I - NUMBER OF COLUMNS (LINES) TO BE SORTED
L - NUMBER OF ROWS (LENGTH OF LINE) PER COLUMN
TEM - TEMPORARY WORK ARRAY OF DIMENSION 'I'
PT - TEMPORARY WORK ARRAY OF DIMENSION 'I'
COL - TEMPORARY WORK ARRAY OF LENGTH 'L'
KEY - IF PRESENT, IS ARRAY OF LENGTH 'L' LISTING THE SORT
KEYS:
KEY(1)=5 IMPLIES THAT THE PRIMARY SORT KEY IS ROW 5
KEY(2)=7 " " " SECONDARY " " " ROW 7
...
KEY(N)=M " " " N-TH " " " ROW M
KEY(N)=0 IMPLIES THAT THE SORT ENDS AFTER N-1 SORT
KEYS ARE USED
TRANA - IF PRESENT, I 63-WORD ARRAY DEFINING THE COLLATING
SEQUENCE.
IF ABSENT OR 0, DISPLAY CODE VALUES ARE USED.
IF 0, KEYM CAN BE USED WITHOUT CHANGING THE
COLLATING SEQUENCE.
KEYM - IF PRESENT, IS AN ARRAY OF LENGTH 'L' FURTHER
DEFINING THE SORT KEYS. (E.G., KEYM(2) IS A MASK
DEFINING WHAT BITS OF THE SECONDARY SORT KEY WILL
BE USED.)

CM REQUIRED: 446B

SUBPROGRAMS REQUIRED

PART OF LANGUAGE

IABS LOCF SHIFT

OTHERS

EQU60

SENT

SSORTL

AUTHOR

C FLINK - KPS - NWL

DATE WRITTEN: 03/08/71

DATE(S) REVISED

06/23/72 - C FLINK

03/23/83 - RECOMPILE AT NOS/BE LEVEL 552

LOCATION OF DECKS

SOURCE

UPDATE LIBRARY ON MSS: NSRDCPL,UN=CSYS

OBJECT

EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC,,ASORT,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'ASORTMV'

PURPOSE

SORT AN ARRAY TAKING ADVANTAGE OF A FAST ARRAY MOVE

FUNCTIONAL CATEGORIES: M1

LANGUAGE: FORTRAN IV

COMPUTERS (OPERATING SYSTEMS)

CDC 6000/CYBER 170 (NOS/BE)

REMARKS

IN ORDER TO USE 'MOVECM', ALL RELATED DATA TO BE SWAPPED
MUST BE PHYSICALLY LOCATED NEXT TO EACH OTHER, THAT IS, EACH
ROW OF 'A' CONTAINS RELATED DATA.

USAGE

CALL ASORTMV (A, NROW, NCOL, IROW, UPDOWN, TEMP, SWAP)

DESCRIPTION OF PARAMETERS

A - 2-DIMENSIONAL ARRAY TO BE SORTED
NROW - NUMBER OF ROWS IN ARRAY 'A' (FIRST DIMENSION)
NCOL - NUMBER OF COLUMNS IN ARRAY 'A' (SECOND DIMENSION)
IROW - ROW POSITION TO BE SORTED
UPDOWN - SORT ORDER DESIRED
1LA - ASCENDING SORT
1LD - DESCENDING SORT
TEMP - WORK ARRAY OF DIMENSION 'NROW' OR GREATER
SWAP - RETURN CODE
0 - NO SWAPPING WAS NECESSARY
(ARRAY ALREADY IN ORDER)
1 - AT LEAST 1 SWAP WAS NECESSARY
2 - UPDOWN INVALID, ASCENDING SORT ASSUMED,
NO SWAPPING WAS NECESSARY
3 - UPDOWN INVALID, ASCENDING SORT ASSUMED,
AT LEAST 1 SWAP WAS NECESSARY
4 - IROW <= 0
5 - IROW > NROW

CM REQUIRED: 257B

SUBPROGRAMS REQUIRED

PART OF LANGUAGE

NONE

OTHERS

MOVECM - MOVE AN ARRAY

AUTHOR

DAVID V SOMMER - DTNSRDC CODE 1892.2

DATE WRITTEN: 02/01/75

DATE(S) REVISED

02/21/80 - CHANGE 'MOVLEV' TO 'MOVECM'

02/15/83 - RECOMPILE AT NOS/BE LEVEL 552

LOCATION OF DECKS

SOURCE

UPDATE LIBRARY ON MSS: NSRDCPL,UN=CSYS

OBJECT

EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC,,ASORTMV,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'BANR'

PURPOSE

PRINT A BANNER (LETTERS ARE 10 LINES HIGH, LINES ARE 110 CHARACTERS LONG)

FUNCTIONAL CATEGORIES: J4 Q0

LANGUAGE: FORTRAN IV

COMPUTERS (OPERATING SYSTEMS)
CDC 6000/CYBER 170 (NOS/BE)

REMARKS

UPPER CASE ONLY (A-Z 0-9 + - * / () \$ = SPACE , . #
[] : " _ ! & ' ? < > @ \ ^ ;)

EACH BANNER REQUIRED 14 LINES (4 BLANKS, 10 FOR THE BANNER).
THUS, 3 BANNERS WILL FIT ON A PAGE AT 6 LINES PER INCH;
5 AT 8 LPI.

UP TO 10 CHARACTERS MAY APPEAR IN A BANNER. THE LINES ARE
110 CHARACTERS LONG.

SEE SUBROUTINE 'BANR6'.

USAGE

CALL BANR (BANNER, IFILE, NEWPAG)

DESCRIPTION OF PARAMETERS

BANNER - 1-10 CHARACTERS TO BE PRINTED
(SINGLE WORD OR ARRAY ELEMENT)

IFILE - NUMBER OF FILE ON WHICH BANNER IS TO BE WRITTEN

NEWPAG - ONE OF:

ZERO - BANNER IS WRITTEN ON NEW PAGE
NON-ZERO - BANNER IS WRITTEN ON SAME PAGE

CM REQUIRED: 1532B

OUTPUT UNITS

UNIT #	LPN/INT	USE
--------	---------	-----

USER SPECIFIES	LJSTABLE OUTPUT	
----------------	-----------------	--

EXAMPLES

PRINT THE BANNER 'HYSTERICAL' AT THE TOP OF THE NEXT PAGE
ON THE PRINTER FILE:

CALL BANR ("HYSTERICAL", 6LOUTPUT, 0)

PRINT THE BANNER '10/19/77' ON THE SAME PAGE ON FILE 9:

CALL BANR (8H10/19/77, 9, 1)

SUBPROGRAMS REQUIRED

PART OF LANGUAGE

SHIFT

OTHERS

VFILL - FILL ARRAY WITH WORD

AUTHOR

DAVID V SOMMER - DTNSRDC CODE 1892.2

DATE WRITTEN: 02/18/75

DATE(S) REVISED

79/07/16 - RE-WRITTEN FOR B7700

81/01/15 - CDC VERSION UPGRADED TO NOS/BE LEVEL 461

83/03/21 - RECOMPILE AT NOS/BE LEVEL 552

LOCATION OF DECKS

SOURCE

UPDATE LIBRARY ON MSS: NSRDCPL,UN=CSYS

OBJECT

EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC,,BANR,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'BANR6'

PURPOSE

PRINT A BANNER (LETTERS ARE 6 LINES HIGH, LINES ARE 80 CHARACTERS LONG)

FUNCTIONAL CATEGORIES: J4 Q0

LANGUAGE: FORTRAN IV

COMPUTERS (OPERATING SYSTEMS)
CDC 6000/CYBER 170 (NOS/BE)

REMARKS

UPPER CASE ONLY (A-Z 0-9 + - * / () \$ = SPACE , . #
[] : " _ ! & ' ? < > @ \ ^ ;)

EACH BANNER REQUIRED 10 LINES (4 BLANKS, 6 FOR THE BANNER).
THUS, 6 BANNERS WILL FIT ON A PAGE AT 6 LINES PER INCH;
8 AT 8 LPI.

UP TO 10 CHARACTERS MAY APPEAR IN A BANNER. THE LINES ARE
80 CHARACTERS LONG.

SEE SUBROUTINE 'BANR'.

USAGE

CALL BANR6 (BANNER, IFILE, NEWPAG)

DESCRIPTION OF PARAMETERS

BANNER - 1-10 CHARACTERS TO BE PRINTED
(SINGLE WORD OR ARRAY ELEMENT)
IFILE - NUMBER OF FILE ON WHICH BANNER IS TO BE WRITTEN
NEWPAG - ONE OF:
ZERO - BANNER IS WRITTEN ON NEW PAGE
NON-ZERO - BANNER IS WRITTEN ON SAME PAGE

CM REQUIRED: 746B

OUTPUT UNITS

UNIT #	LFN/INT	USE

USER SPECIFIES		LISTABLE OUTPUT

EXAMPLES

PRINT THE BANNER 'HYSTERICAL' AT THE TOP OF THE NEXT PAGE
ON THE PRINTER FILE:

CALL BANR6 ("HYSTERICAL", 6LOUTPUT, 0)

PRINT THE BANNER '10/19/77' ON THE SAME PAGE ON FILE 9:

CALL BANR6 (8H10/19/77, 9, 1)

SUBPROGRAMS REQUIRED
PART OF LANGUAGE
MOVLEV SHIFT
OTHERS
NONE

AUTHOR
DAVID V SOMMER - DTNSRDC CODE 1892.2

DATE WRITTEN: 10/18/77

DATE(S) REVISED
79/07/16 - RE-WRITTEN FOR B7700
83/03/23 - RECOMPILE AT NOS/BE LEVEL 552

LOCATION OF DECKS
SOURCE
UPDATE LIBRARY ON MSS: NSRDCPL,UN=CSYS
OBJECT
EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT
BEGIN,DOCGET,,NSRDC,,BANR6,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'BESSI'

PURPOSE

MODIFIED BESSEL FUNCTION OF THE FIRST KIND

FUNCTIONAL CATEGORIES: C3

LANGUAGE: FORTRAN IV

COMPUTERS (OPERATING SYSTEMS)

CDC 6000/CYBER 170 (NOS/BE)

REMARKS

FOR $N=0$, $I(NU)$ AND $I(NU+1)$ ARE ALWAYS COMPUTED.

IF K-BESSEL FUNCTION IS ALSO REQUIRED, USE SUBROUTINE
BESSK TO OBTAIN I_- AND K-BESSEL FUNCTIONS.

USAGE

CALL BESSI (X, FNU, N, VI)

DESCRIPTION OF PARAMETERS

X - THE ARGUMENT ($X > 0.0$)
FNU - NU, THE FRACTIONAL PART OF THE ORDER ($0. \leq FNU \leq 1.$)
N - HIGHEST ORDER IS $(N+FNU)$
ABS(N)+1 TABLE ENTRIES ARE TO BE COMPUTED
VI - ARRAY TO CONTAIN THE COMPUTED TABLE
(DIMENSION MUST BE AT LEAST: $MAX(N+13, X+28)$, THE
REST OF THE ARRAY IS WORK AREA)
 $VI(1) = (E^{*-X}) I_0(X)$, WHERE I_0 IS $I(0+FNU)$
ETC.

CM REQUIRED: 640B

METHOD

SEE "RECURRENCE TECHNIQUES FOR THE CALCULATION OF BESSEL
FUNCTIONS", M. GOLDSTEIN AND R. THALER, MTAC, VOL. XIII,
NO. 66, APRIL 1959.

FOR $X \geq 10.0$, ASYMPTOTIC VALUES ARE COMPUTED USING THE
SO-CALLED PHASE AMPLITUDE METHOD. SEE "BESSEL FUNCTIONS FOR
LARGE ARGUMENTS", M. GOLDSTEIN AND R. THALER, MTAC, VOL XII,
NO. 61, JANUARY 1958.

SUBPROGRAMS REQUIRED

PART OF LANGUAGE

ABS EXP IABS MAX0 SQRT

OTHERS

GAMMA

AUTHORS

FLORENCE F. RAGUSA AND M. GOLDSTEIN
HARVEY ABRAMSON
MARGARET FRANTZ
NEW YORK UNIVERSITY

VIM ROUTINE NYUBESS

DATE WRITTEN: BEFORE 11/65

DATE(S) REVISED

11/65 - HA
09/01/67 - MF
09/20/83 - RECOMPILE AT NOS/BE LEVEL 552

LOCATION OF DECKS

SOURCE

UPDATE LIBRARY ON MSS: NSRDCPM,UN=CSYS

OBJECT

EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC,,BESSI,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'BESSJ'

PURPOSE

BESSEL FUNCTION OF THE FIRST KIND

FUNCTIONAL CATEGORIES: C3

LANGUAGE: FORTRAN IV

COMPUTERS (OPERATING SYSTEMS)

CDC 6000/CYBER 170 (NOS/BE)

REMARKS

FOR $N=0$, $J(NU)$ AND $J(NU+1)$ ARE ALWAYS COMPUTED.

IF Y-BESSEL FUNCTION IS ALSO REQUIRED, USE SUBROUTINE
BESSY TO OBTAIN J- AND Y-BESSEL FUNCTIONS.

USAGE

CALL BESSJ (X, FNU, N, VJ)

DESCRIPTION OF PARAMETERS

X - THE ARGUMENT ($X > 0.0$)
FNU - NU, THE FRACTIONAL PART OF THE ORDER ($0. \leq FNU \leq 1.$)
N - HIGHEST ORDER IS $(N+FNU)$
ABS(N)+1 TABLE ENTRIES ARE TO BE COMPUTED
VI - ARRAY TO CONTAIN THE COMPUTED TABLE
(DIMENSION MUST BE AT LEAST: $\text{MAX}(N+13, X+28)$, THE
REST OF THE ARRAY IS WORK AREA)
VJ(1) = $J_0(X)$, WHERE J_0 IS $J(0+FNU)$
ETC.

CM REQUIRED: 642B

METHOD

SEE "RECURRENCE TECHNIQUES FOR THE CALCULATION OF BESSEL
FUNCTIONS", M. GOLDSTEIN AND R. THALER, MTAC, VOL. XIII,
NO. 66, APRIL 1959.

FOR $X \geq 10.0$, ASYMPTOTIC VALUES ARE COMPUTED USING THE
SO-CALLED PHASE AMPLITUDE METHOD. SEE "BESSEL FUNCTIONS FOR
LARGE ARGUMENTS", M. GOLDSTEIN AND R. THALER, MTAC, VOL XII,
NO. 61, JANUARY 1958.

SUBPROGRAMS REQUIRED

PART OF LANGUAGE

ABS COS IABS MAX0 SIN
SQRT

OTHERS

GAMMA

AUTHORS

FLORENCE F. RAGUSA AND M. GOLDSTEIN
HARVEY ABRAMSON
MARGARET FRANTZ
NEW YORK UNIVERSITY

VIM ROUTINE NYUBESS

DATE WRITTEN: BEFORE 11/65

DATE(S) REVISED

11/65 - HA
09/01/67 - MF
09/20/83 - RECOMPILE AT NOS/BE LEVEL 552

LOCATION OF DECKS

SOURCE

UPDATE LIBRARY ON MSS: NSRDCPM,UN=CSYS

OBJECT

EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC,,BESSJ,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'BESSK'

PURPOSE

BESSEL FUNCTION OF THE SECOND KIND

FUNCTIONAL CATEGORIES: C3

LANGUAGE: FORTRAN IV

COMPUTERS (OPERATING SYSTEMS)
CDC 6000/CYBER 170 (NOS/BE)

REMARKS

FOR $N=0$, $I(NU)$ AND $I(NU+1)$ ARE ALWAYS COMPUTED.

THIS SUBROUTINE ALSO COMPUTED THE I-BESSEL FUNCTION.

USAGE

CALL BESSK (X, FNU, N, VI, VK)

DESCRIPTION OF PARAMETERS

X - THE ARGUMENT ($X > 0.0$)
FNU - NU, THE FRACTIONAL PART OF THE ORDER ($0. \leq FNU \leq 1.$)
N - HIGHEST ORDER IS $(N+FNU)$
ABS(N)+1 TABLE ENTRIES ARE TO BE COMPUTED
VI - ARRAY TO CONTAIN THE COMPUTED TABLE
(DIMENSION MUST BE AT LEAST: $\text{MAX}(N+13, X+28)$, THE
REST OF THE ARRAY IS WORK AREA)
 $VI(1) = (E^{**(-X)}) I_0(X)$, WHERE I_0 IS $I(0+FNU)$
ETC.
VK - ARRAY TO CONTAIN THE COMPUTED K-TABLE
(DIMENSION MUST BE AT LEAST: $\text{MAX}(N+13, X+28)$, THE
REST OF THE ARRAY IS WORK AREA)
 $VK(1) = (E^{**(X)}) I_0(X)$, WHERE I_0 IS $I(0+FNU)$
ETC.

CM REQUIRED: 437B

METHOD

SEE "RECURRENCE TECHNIQUES FOR THE CALCULATION OF BESSEL
FUNCTIONS", M. GOLDSTEIN AND R. THALER, MTAC, VOL. XIII,
NO. 66, APRIL 1959.

FOR $X \geq 10.0$, ASYMPTOTIC VALUES ARE COMPUTED USING THE
SO-CALLED PHASE AMPLITUDE METHOD. SEE "BESSEL FUNCTIONS FOR
LARGE ARGUMENTS", M. GOLDSTEIN AND R. THALER, MTAC, VOL XII,
NO. 61, JANUARY 1958.

SUBPROGRAMS REQUIRED

PART OF LANGUAGE
A LOG EXP IABS MAX0 SIN
OTHERS
BESSI
GAMMA

AUTHORS

FLORENCE F. RAGUSA AND M. GOLDSTEIN
HARVEY ABRAMSON
MARGARET FRANTZ
NEW YORK UNIVERSITY

VIM ROUTINE NYUBESS

DATE WRITTEN: BEFORE 11/65

DATE(S) REVISED

11/65 - HA
09/01/67 - MF
09/20/83 - RECOMPILE AT NOS/BE LEVEL 552

LOCATION OF DECKS

SOURCE

UPDATE LIBRARY ON MSS: NSRDCPM,UN=CSYS

OBJECT

EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC,,BESSK,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'BESSY'

PURPOSE

BESSEL FUNCTION OF THE SECOND KIND

FUNCTIONAL CATEGORIES: C3

LANGUAGE: FORTRAN IV

COMPUTERS (OPERATING SYSTEMS)

CDC 6000/CYBER 170 (NOS/BE)

REMARKS

FOR $N=0$, $Y(NU)$ AND $Y(NU+1)$ ARE ALWAYS COMPUTED.

THIS SUBROUTINE ALSO COMPUTES THE J-BESSEL FUNCTION.

USAGE

CALL BESSY (X, FNU, N, VJ, VY)

DESCRIPTION OF PARAMETERS

X - THE ARGUMENT ($X > 0.0$)

FNU - NU, THE FRACTIONAL PART OF THE ORDER ($0. \leq FNU \leq 1.$)

N - HIGHEST ORDER IS $(N+FNU)$

ABS(N)+1 TABLE ENTRIES ARE TO BE COMPUTED

VJ - ARRAY TO CONTAIN THE COMPUTED TABLE

(DIMENSION MUST BE AT LEAST: $\text{MAX}(N+13, X+28)$, THE
REST OF THE ARRAY IS WORK AREA)

$VJ(1) = J_0(X)$, WHERE J_0 IS $J(0+FNU)$

ETC.

VY - ARRAY TO CONTAIN THE COMPUTED Y-TABLE

(DIMENSION MUST BE AT LEAST: $\text{MAX}(N+13, X+28)$, THE
REST OF THE ARRAY IS WORK AREA)

$VY(1) = Y_0(X)$, WHERE Y_0 IS $Y(0+FNU)$

ETC.

CM REQUIRED: 416B

METHOD

SEE "RECURRENCE TECHNIQUES FOR THE CALCULATION OF BESSEL
FUNCTIONS", M. GOLDSTEIN AND R. THALER, MTAC, VOL. XIII,
NO. 66, APRIL 1959.

FOR $X \geq 10.0$, ASYMPTOTIC VALUES ARE COMPUTED USING THE
SO-CALLED PHASE AMPLITUDE METHOD. SEE "BESSEL FUNCTIONS FOR
LARGE ARGUMENTS", M. GOLDSTEIN AND R. THALER, MTAC, VOL XII,
NO. 61, JANUARY 1958.

SUBPROGRAMS REQUIRED

PART OF LANGUAGE

ABS

COS

IABS

MAX0

SIN

SQRT

OTHERS

BESSJ

GAMMA

AUTHORS

FLORENCE F. RAGUSA AND M. GOLDSTEIN
HARVEY ABRAMSON
MARGARET FRANTZ
NEW YORK UNIVERSITY

VIM ROUTINE NYUBESS

DATE WRITTEN: BEFORE 11/65

DATE(S) REVISED

11/65 - HA
09/01/67 - MF
09/20/83 - RECOMPILE AT NOS/BE LEVEL 552

LOCATION OF DECKS

SOURCE

UPDATE LIBRARY ON MSS: NSRDCPM,UN=CSYS

OBJECT

EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC,,BESSY,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'BSJ'

PURPOSE

SPHERICAL BESSEL FUNCTION

FUNCTIONAL CATEGORIES: C3

LANGUAGE: FORTRAN IV

COMPUTERS (OPERATING SYSTEMS)

CDC 6000/CYBER 170 (NOS/BE)

REMARKS

EVALUATES THE SPHERICAL BESSEL FUNCTION J-SUB-N(X) FOR
N=-1,0,...,I BY MEANS OF A RECURSIVE RELATION AND REASONABLE
STARTING VALUES. STARTING VALUES ARE GENERATED WITHIN THE
SUBROUTINE.

USAGE

CALL BSJ (I, X, BJ)

DESCRIPTION OF PARAMETERS

I - HIGHEST ORDER DESIRED
X - SINGLE PRECISION FLOATING POINT VARIABLE
BJ - ARRAY DIMENSIONED AT LEAST I+2 FOR SOLUTIONS
(BJ(N+2) = J-SUB-N(X))

CM REQUIRED: 404B

METHOD

A. THE VALUES ARE COMPUTED BY USING THE RECURSION FORMULA:

$$J_{I-1}(X) + J_{I+1}(X) = \frac{2I+1}{X} J_I(X)$$

IF $X > 20.5$, THE RECURSION IS FORWARD.

IF $X < 20.5$, THE RECURSION IS BACKWARD.

FOR VARIOUS RANGES ($X < 20.5$), AN UPPER LIMIT, NU, IS SET.

BJ(NU+1) IS THEN SET TO ZERO, AND THE RECURSION PROCESS
IS EXECUTED.

B. RANGE: THE FOLLOWING DOMAINS HAVE BEEN CAREFULLY CHECKED:

$1 < X < 25$; $I < 25$. ERROR IS LESS THAN $+5 \times 10^{-11}$. POSSIBLE
DOMAINS ARE: $0 < I < 25$ AND $0 < X < 100$. (CAUTION: FOR LARGER
DOMAINS, CHECK DIMENSIONING IN THE SUBROUTINE.)

NOTE: IF $I \gg X$, J-SUB-I(X) IS VERY SMALL.

REFERENCES

HANDBOOK OF MATHEMATICAL FUNCTIONS, AMS 55, NATIONAL BUREAU
OF STANDARDS.

ASSOCIATION OF COMPUTING MACHINERY, "GENERATION OF SPHERICAL
BESSEL FUNCTIONS", F. J. CORBATO AND J. L. URETSKY, JULY
1959, VOL. 6, NO. 3, PP. 366-375.

SUBPROGRAMS REQUIRED

PART OF LANGUAGE

COS SIN

OTHERS

NONE

AUTHORS

R L PEXTON - LAWRENCE RADIATION LABORATORY

D A WILBER - LAWRENCE RADIATION LABORATORY

DATE WRITTEN: 01/06/65 (RLP)

DATE(S) REVISED

11/65 (DAW)

09/20/83 - RECOMPILE AT NOS/BE LEVEL 552

LOCATION OF DECKS

SOURCE

UPDATE LIBRARY ON MSS: NSRDCPM,UN=CSYS

OBJECT

EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC,,BSJ,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'BUFSIZE'

PURPOSE

PRINT MESSAGE IN DAYFILE FOR EACH FILE SPECIFIED
INDICATING BUFFER SIZE AND WHETHER BUFFER IS
CURRENTLY ALLOCATED.

FUNCTIONAL CATEGORIES: QO

LANGUAGE: CDC 6000 CP COMPASS

COMPUTERS (OPERATING SYSTEMS)
CDC 6000/CYBER 170 (NOS/BE)

REMARKS

THIS ROUTINE PICKS UP THE BUFFER SIZE (BFS) FROM WORD 4 OF
THE FIT. IT ALSO DETERMINES IF THE BUFFER IS CURRENTLY
ALLOCATED BY CHECKING THE BUFFER FIRST WORD ADDRESS (FWB)
IN WORD 6 OF THE FIT. IT PRINTS A REPORT IN THE DAYFILE
OF THE FORM:

	FILE BUFFER SIZES	
FILE (LFN)	SIZE (OCTAL)	ALLOCATED
XXXXXXX	XXXXXX	Y OR N

USAGE

CALLED FROM COBOL PROGRAM
ENTER BUFSIZE USING FILENAME1, FILENAM2,.....
WHERE FILENAMEX IS NAME OF FILE IN FD STATEMENT

CALLED FROM FTN PROGRAM
CALL BUFSIZE (FIT1, FIT2,)
WHERE FITX IS ADDRESS OF A FILE INFORMATION
TABLE.

CM REQUIRED: 72B WORDS

SUBPROGRAMS REQUIRED

PART OF LANGUAGE
NONE
OTHERS
NONE

AUTHOR

BRUCE D. BLACK - DTNSRDC CODE 1892.1 (CDC)

DATE WRITTEN: 04/07/78

DATE(S) REVISED

LOCATION OF DECKS

SOURCE
UPDATE LIBRARY ON MSS: NSRDCPL,UN=CSYS
OBJECT
EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC,,BUFSIZE,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'CELLI'
SUBROUTINE 'ELLI'

PURPOSE

COMPLETE AND INCOMPLETE ELLIPTIC INTEGRALS OF THE FIRST AND
SECOND KIND

FUNCTIONAL CATEGORIES: C3

LANGUAGE: FORTRAN IV EXTENDED

COMPUTERS (OPERATING SYSTEMS)
CDC 6000/CYBER 170 (NOS/BE)

REMARKS

CELLI IS AN ENTRY POINT IN ELLI.

WHEN $ABS(PHI) \leq PI/2$, F AND E ARE ACCURATE TO AT LEAST 10
SIGNIFICANT FIGURES. AS $ABS(PHI)$ GETS LARGE, THE ACCURACY
WILL NOT BE AS GOOD SINCE ELLI USES THE TANGENT SUBROUTINE
WHICH BECOMES LESS ACCURATE AS THE ANGLE $ABS(PHI)$ INCREASES.

USAGE

CALL CELLI (PHI, CAY, F, E)
CALL ELLI (PHI, CAY, F, E)

DESCRIPTION OF PARAMETERS

PHI - UPPER LIMIT OF INTEGRAL
(NOT USED BY CELLI WHICH ASSUMES $PI/2$)
CAY - THE PARAMETER IN THE INTEGRAL
F - OUTPUT THE ELLIPTIC INTEGRAL OF THE FIRST KIND
(F(PHI,CAY))
E - OUTPUT THE ELLIPTIC INTEGRAL OF THE SECOND KIND
(E(PHI,CAY))

CM REQUIRED: 351B

ERROR MESSAGES

IF $K > 1$, F AND E DO NOT EXIST. A MESSAGE IS PRINTED AND
F AND E ARE SET TO PHI.

IF $K=1$ AND $ABS(PHI) \geq PI/2$, F DOES NOT EXIST. A MESSAGE IS
PRINTED AND F IS SET TO $SIGN(PHI)*1.0E+294$. E EXISTS AND IS
COMPUTED.

OUTPUT UNITS

UNIT #	LFN	USE
-----	-----	-----
	OUTPUT	ERROR MESSAGES PRINTED BY LABRT

METHOD

WHEN $K < 1$, LANDEN'S TRANSFORMATION IS USED.

WHEN $K=1$, E IS COMPUTED BY:

$$E(\text{PHI},1) = N + \text{ABS}(\text{SIN}(\text{PHI}) - \text{SIN}(N*\text{PI}/2))$$

WHERE N IS THE INTEGRAL PART OF $\text{PHI}*(2/\text{PI})$.

WHEN $K=1$ AND $\text{ABS}(\text{PHI}) < \text{PI}/2$, F IS COMPUTED BY:

$$F(\text{PHI},1) = .5 * \text{LN} \left(\frac{1+\text{SIN}(\text{PHI})}{1-\text{SIN}(\text{PHI})} \right)$$

REFERENCE: "HANDBOOK OF MATHEMATICAL FUNCTIONS" BY M.
ABRAMOWITZ AND I. A. STEGUN.

SUBPROGRAMS REQUIRED

PART OF LANGUAGE

ABS	AIN	ALOG	AMIN1	AMOD
ATAN	FLOAT	INT	MOD	SIGN
SIN	SQRT	TAN		

OTHERS

LABRT - PRINT ERROR MESSAGES (57B)

AUTHORS

KARL J MELENDEZ
DUANE HARDER
LOS ALAMOS SCIENTIFIC LABORATORY

VIM ROUTINE LASL C304A

DATE WRITTEN: 02/05/68

DATE(S) REVISED

02/69 - DH
09/20/83 - RECOMPILE AT NOS/BE LEVEL 552

LOCATION OF DECKS

SOURCE

UPDATE LIBRARY ON MSS: NSRDCPM,UN=CSYS (*DECK ELLI)

OBJECT

EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC,,CELLI,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'CENTER'

PURPOSE

CENTER A CHARACTER STRING WITHIN AN OUTPUT FIELD

FUNCTIONAL CATEGORIES: M4

LANGUAGE: FORTRAN IV

COMPUTERS (OPERATING SYSTEMS)

CDC 6000/CYBER 170 (NOS/BE)

USAGE

CALL CENTER (IN, LIN, OUT, NOUTC)

DESCRIPTION OF PARAMETERS

IN - INPUT ARRAY CONTAINING CHARACTER STRING TO BE
CENTERED
(CHARACTER STRING STARTS IN POSITION 1 AND ENDS
WITH LAST NON-BLANK CHARACTER)
LIN - NUMBER OF WORDS IN 'IN'
OUT - OUTPUT ARRAY IN WHICH 'IN' IS TO BE CENTERED
NOUTC - NUMBER OF CHARACTERS IN 'OUT' WITHIN WHICH
'IN' IS TO BE CENTERED

REMARKS

USEFUL FOR CENTERING HEADINGS ON A PAGE. FOR INSTANCE,
IF 'THIS IS A HEADING' IS TO BE CENTERED FOR A 132-
COLUMN WIDE PAGE, THE FOLLOWING CAN BE USED:

INTEGER IN(2), OUT(14)

IN(1) = 10HTHIS IS A

IN(2) = 10HHEADING

CALL CENTER (IN, 2, OUT, 132)

ON RETURN, 'OUT' WILL CONTAIN 'THIS IS A HEADING' IN
POSITIONS 58 THRU 74 (WORD 6, POSITION 8 THRU
WORD 8, POSITION 4). POSITIONS 1-56 AND 75-132 WILL
CONTAIN BLANKS.

SUBPROGRAMS REQUIRED

PART OF LANGUAGE

MOD

OTHERS

GETCHA - GET CHARACTER FROM ARRAY

LASTC - FIND LAST NON-BLANK IN ARRAY

MOVECM - FAST ARRAY MOVE

PUTCHA - INSERT CHARACTER INTO ARRAY

VFILL - FILL ARRAY WITH SPECIFIED WORD

CM REQUIRED: 144B

AUTHOR

DAVID V SOMMER - DTNSRDC CODE 1892.2

DATE WRITTEN: 05/19/75

DATE(S) REVISED

05/04/76

03/16/78

04/03/83 - CHANGE MOVLEV TO MOVECM

LOCATION OF DECKS

SOURCE

UPDATE LIBRARY ON MSS: NSRDCPL,UN=CSYS

OBJECT

EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC,,CENTER,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'CGAUSS'

PURPOSE

COMPLEX SOLUTION OF SIMULTANEOUS EQUATIONS AND DETERMINANT
BY ITERATIVE GAUSSIAN ELIMINATION

FUNCTIONAL CATEGORIES: F4

LANGUAGE: FORTRAN IV EXTENDED

COMPUTERS -(OPERATING SYSTEMS)
CDC 6000/CYBER 170 (NOS/BE)

REMARKS

TO INCREASE BEYOND 10 BY 10, THE DIMENSIONS OF ARRAYS A, AA,
B, BB, X, XX AND IN MUST BE CHANGED.

USAGE

COMPLEX AA(10,10), BB(10,10), XX(10,10), DET
CALL CGAUSS (N, M, AA, BB, XX, VAL2, DET, MM, NA, NX)

DESCRIPTION OF PARAMETERS

N - NUMBER OF ROWS OF AA, BB, XX (MAX: 10)
M - NUMBER OF COLUMNS OF RIGHT-HAND SIDES (MAX: 10)
AA - COMPLEX ARRAY OF COEFFICIENTS FOR SIMULTANEOUS
EQUATIONS $AA*XX=BB$ (MAX: 10 BY 10)
BB - COMPLEX ARRAY OF RIGHT-HAND-SIDES FOR $AA*XX=BB$
(MAX: 10 BY 10)
XX - COMPLEX ARRAY OF SOLUTIONS OF $AA*XX=BB$
(MAX: 10 BY NX)
VAL2 - OUTPUT THE INFINITY NORM OF THE CORRECTION
DET - OUTPUT THE COMPLEX DETERMINANT OF AA
MM - NUMBER OF ITERATIONS
(MM=0 RETURNS THE RESULT OF THE FIRST GAUSSIAN
ELIMINATION)
NA - DIMENSIONS OF AA AND BB AND FIRST DIMENSION OF XX
NX - SECOND DIMENSION OF XX

CM REQUIRED: 1721B

METHOD

A FIRST SOLUTION FOR XX IS OBTAINED DIRECTLY. $BB-AA*XX$ IS
CALCULATED AS DD. THE RESIDUAL EQUATION $AA*X=DD$ IS SOLVED
AND THE SOLUTION ADDED TO XX. THIS PROCESS CONTINUES FOR MM
CYCLES. IF MM=0, THE RESULT OF THE FIRST GAUSSIAN
ELIMINATION IS RETURNED.

SUBPROGRAMS REQUIRED
PART OF LANGUAGE
CABS
OTHERS
NONE

AUTHORS
UNIVERSITY OF MARYLAND
SUE VOIGT

DATE WRITTEN: 1971

DATE(S) REVISED
09/20/83 - RECOMPILE AT NOS/BE LEVEL 552

LOCATION OF DECKS
SOURCE
UPDATE LIBRARY ON MSS: NSRDCPM,UN=CSYS (*DECK AMCGAUS)
OBJECT
EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT
BEGIN,DOCGET,,NSRDC,,CGAUSS,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'CHFILL'

PURPOSE

FILL (PORTION OF) AN ARRAY WITH A CHARACTER

FUNCTIONAL CATEGORIES: M4

LANGUAGE: FORTRAN IV EXTENDED

COMPUTERS (OPERATING SYSTEMS)

CDC 6000/CYBER 170 (NOS/BE)

REMARKS

NONE

USAGE

CALL CHFILL (FILLCH, TO, TOPOS, NCHAR)

DESCRIPTION OF PARAMETERS

FILLCH - FILL CHARACTER (1R OR 1H OR " ")

TO - INTEGER ARRAY TO BE FILLED

TOPOS - STARTING CHARACTER POSITION IN 'TO'
(CHARACTER 1 IS LEFT-MOST CHARACTER OF TO(1))

NCHAR - NUMBER OF CHARACTERS TO BE FILLED

CM REQUIRED: 35B

EXAMPLE

TO: *****

AFTER CALL CHFILL (1R/, TO, 23, 7)

TO: *****//*****

SUBPROGRAMS REQUIRED

PART OF LANGUAGE

AND SHIFT

OTHERS

PUTCHA - INSERT CHARACTER INTO ARRAY

ARITHMETIC STATEMENT FUNCTIONS

L11FMT - FAST L-FORMAT DECODE (LEFT-ADJ, ZERO-FILLED)

R11FMT - FAST R-FORMAT DECODE (RIGHT-ADJ, ZERO-FILLED)

AUTHOR

DAVID V SOMMER - DTNSRDC CODE 1892.2

DATE WRITTEN: 03/10/77

DATE(S) REVISED

03/23/83 - RECOMPILE AT NOS/BE LEVEL 552

LOCATION OF DECKS

SOURCE

UPDATE LIBRARY ON MSS: NSRDCPL,UN=CSYS

OBJECT

EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC,,CHFILL,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'CHNGSEQ'

PURPOSE

ALLOW COBOL4 USER TO DEFINE HIS OWN COLLATING SEQUENCE

FUNCTIONAL CATEGORIES: M4

LANGUAGE: CDC 6000 CP COMPASS

COMPUTERS (OPERATING SYSTEMS)

CDC 6000/CYBER 170 (NOS/BE)

REMARKS

USER MUST USE THE U OPTION ON THE COBOL CALL CARD.

BINARY ZERO IS THE 64TH CHARACTER IN THE COLLATING SEQUENCE.

ROUTINE SETS TABLES AFFECTING COLLATING SEQUENCE FOR
COBOL IF TESTS, COBOL SORT, INDEX SEQ FILE SEQUENCE, ETC.

USAGE

CALL CHNGSEQ USING MYTBL.

DESCRIPTION OF PARAMETER

THE USER MUST SET UP A DATA ITEM 63 CHARACTERS IN
LENGTH CONTAINING THE CHARACTERS IN THE ORDER HE
WISHES THE COLLATING SEQUENCE TO BE. ALL 63
CHARACTERS MUST BE PRESENT.

NOTE: TO SET " INTO THE STRING, REDEFINE AND USE
MOVE QUOTE TO

EXAMPLE:

```
01 MYTBL PIC X(63) VALUE " @:[_#&'?>\^.);+$*-/,(<= <ABCDEF  
- "GHI!JKLMNOPQR]STUVWXYZ0123456789".  
01 MYTBLA REDEFINES MYTBL.  
03 ENTR PIC X OCCURS 63 TIMES.
```

RECALL THAT A NON-NUMERIC LITERAL MUST CONTINUE THRU
COL 72 OF THE FIRST CARD AND THAT CONTINUATION CARD
MUST HAVE A HYPHEN IN COL 7. (EXAMPLE HERE DOESN'T
GO TO COL. 72).

PROCEDURE DIVISION.

PAR1.

```
MOVE QUOTE TO ENTR (24).  
CALL CHNGSEQ USING MYTBL.
```

CM REQUIRED: 27B

METHOD

COLLATING SEQUENCE TABLES IN COBOL OBJECT TIME ROUTINES
ARE CHANGED.

SUBPROGRAMS REQUIRED

PART OF LANGUAGE

NONE

OTHERS

NONE

AUTHOR

BRUCE D BLACK DTNSRDC 1892.1 (CDC)

DATE WRITTEN: 11/15/77

DATE(S) REVISED

LOCATION OF DECKS

SOURCE

UPDATE LIBRARY ON MSS: NSRDCPL,UN=CSYS

OBJECT

EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC,,CHNGSEQ,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'CLUNLD'

PURPOSE

CLOSE AND UNLOAD A FILE

FUNCTIONAL CATEGORIES: Q3

LANGUAGE: FORTRAN IV EXTENDED

COMPUTERS (OPERATING SYSTEMS)

CDC 6000/CYBER 170 (NOS/BE)

REMARKS

CAUTION: FOR ICLT=1 OR 2, BE SURE BUFFERS HAVE BEEN FLUSHED BEFORE UNLOADING A PERMANENT FILE IF YOU EXPECT TO USE IT AGAIN. (I.E., CLOSE THE FILE BEFORE CALLING CLUNLD)

CAUTION: RANDOM PERMANENT FILES MUST BE CLOSED BEFORE CLUNLD IS CALLED TO INSURE THAT THE LATEST INDEX IS WRITTEN.

FORTRAN SEQUENTIAL FILES SHOULD HAVE THEIR BUFFERS FLUSHED BE REWINDING THEM BEFORE CALLING CLUNLD.

USAGE

CALL CLUNLD (IERR, ICLT, LFN)

DESCRIPTION OF PARAMETERS

IERR - ERROR RETURN CODE (0=NO ERRORS)

ICLT - TYPE OF CONTENTS OF 'LFN'

1 - LFN CONTAINS THE ADDRESS OF A FET.

A CLOSE-UNLOAD IS PERFORMED ON THIS FET.

2 - LFN CONTAINS AN LFN TO BE UNLOADED.

A DUMMY FET IS CREATED AND THE FILE UNLOADED.

3 - LFN CONTAINS A FILE NAME OR FORTRAN LOGICAL UNIT NUMBER (I.E., ANY FILE ON THE FORTRAN PROGRAM STATEMENT). THE FIT WILL BE FOUND AND THE FILE UNLOADED.

NOTE: CLOSEM (A RECORD MANAGER ROUTINE) IS CALLED TO CLOSE THE FILE.

NOTE: A DUMMY FET IS CREATED TO UNLOAD A FILE THAT RECORD MANAGER DOESN'T KNOW HAS BEEN ATTACHED.

LFN - CONTENTS IS DETERMINED BY 'ICLT'

CM REQUIRED: 22B

SUBPROGRAMS REQUIRED

PART OF LANGUAGE

NONE

OTHERS

CLUXXX FNDFIT INDCMT IZONK ZIO

AUTHOR

C M CHERNICK - DTNSRDC CODE 1832

DATE WRITTEN: 11/15/71

DATE(S) REVISED

06/01/72 11/20/74 02/21/75 05/75 04/76

LOCATION OF DECKS

SOURCE

CODE 1832

OBJECT

EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC,,CLUNLD,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'CMPINV'

PURPOSE

COMPLEX MATRIX INVERSION

FUNCTIONAL CATEGORIES: F4 A2

LANGUAGE: FORTRAN IV EXTENDED

COMPUTERS (OPERATING SYSTEMS)

CDC 6000/CYBER 170 (NOS/BE)

REMARKS

THE REAL AND/OR IMAGINARY PARTS OF THE MATRIX 'A' MAY BE SINGULAR.

USAGE

CALL CMPINV (A, N, N1, C, ID, E, N2, INDEX)

DESCRIPTION OF PARAMETERS

A - COMPLEX INPUT MATRIX
(NOT DESTROYED BY SUBROUTINE)
N - DIMENSION OF A AND C (N X N)
N1 - NUMBER OF ROWS IN A AND C CURRENTLY FULL
C - INVERSE RESULT MATRIX
(MAY BE THE SAME AS A)
ID - RETURN CODE
1 - INVERSION SUCCESSFUL
2 - MATRIX SINGULAR
E - TEMPORARY ARRAY SOLVING N2 X N2 SYSTEM
N2 - NO SMALLER THAN N1+N1
INDEX - TEMPORARY ARRAY USED IN INVERSION (N2,3)

THE CALLING PROGRAM MUST INCLUDE:

COMPLEX A(N,N), C(N,N)
REAL E(N2,N2), INDEX(N2,3)

CM REQUIRED: 134B

METHOD

THE SYSTEM SOLVED IS THE EXPANDED MATRIX

$$E = \begin{bmatrix} \text{!} & \text{!} & \text{AR} & \text{!} & \text{!} \\ \text{!} & \text{!} & & & \\ \text{!} & \text{!} & \text{AI} & & \\ \text{!} & \text{!} & & \text{AR} & \text{!} \end{bmatrix}$$

WHERE CR IS TAKEN AS THE UPPER LEFT CORNER OF THE INVERSE AND CI IS TAKEN AS THE LOWER LEFT CORNER OF THE INVERSE. (LANCZOS, APPLIED ANALYSIS, P 137). THE INVERSE IS COMPUTED BY SUBROUTINE MATINS (ALSO ON NSRDC) WHICH USES GAUSS-JORDAN ELIMINATION. THIS METHOD FINDS AN INVERSE IF IT EXISTS, EVEN IF REAL AND IMAGINARY PARTS OF A ARE BOTH INDIVIDUALLY SINGULAR. IDENTIFICATION OF A SINGULAR COMPLEX MATRIX IS RETURNED TO THE CALLING PROGRAM.

SUBPROGRAMS REQUIRED
PART OF LANGUAGE
 AIMAG CMPLX REAL
OTHERS
 MATINS - MATRIX INVERSION

AUTHOR
 SHARON E GOOD - DTNSRDC CODE 1892.1

DATE WRITTEN: 06/10/71

DATE(S) REVISED
 09/20/83 - RECOMPILE AT NOS/BE LEVEL 552

LOCATION OF DECKS
SOURCE
 UPDATE LIBRARY ON MSS: NSRDCPM,UN=CSYS (*DECK AMCMAT)
OBJECT
 EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT
 BEGIN,DOCGET,,NSRDC,,CMPINV,OUTPUT,MSACCES=<PASSWORD>.

FUNCTION 'COMPSTR'

PURPOSE

COMPARE TWO CHARACTER STRINGS

FUNCTIONAL CATEGORIES: MO

LANGUAGE: FORTRAN IV EXTENDED

COMPUTERS (OPERATING SYSTEMS)

CDC 6000/CYBER 170 (NOS/BE)

REMARKS

NONE

USAGE

TEST = COMPSTR (A, FROMA, B, FROMB, NCHAR)

DESCRIPTION OF PARAMETERS

A - ARRAY CONTAINING FIRST CHARACTER STRING
FROMA - STARTING CHARACTER POSITION IN A
(POSITION 1 IS LEFT-MOST 6-BIT CHARACTER IN A(1))
B - ARRAY CONTAINING SECOND CHARACTER STRING
FROMB - STARTING CHARACTER POSITION IN B
(POSITION 1 IS LEFT-MOST 6-BIT CHARACTER IN B(1))
NCHAR - NUMBER OF CHARACTERS TO COMPARE
COMPSTR - WILL RETURN ONE OF:
-1. - STRING IN A IS LESS THAN STRING IN B
0. - STRING IN A IS EQUAL TO STRING IN B
+1. - STRING IN A IS GREATER THAN STRING IN B

CM REQUIRED: 66B

SUBPROGRAMS REQUIRED

PART OF LANGUAGE

NONE

OTHERS

GETCHA - GET CHARACTER FROM ARRAY

AUTHOR

DAVID V SOMMER - DTNSRDC CODE 1892.2

DATE WRITTEN: 04/04/77

DATE(S) REVISED

03/23/83 - RECOMPILE AT NOS/BE LEVEL 552

LOCATION OF DECKS

SOURCE

UPDATE LIBRARY ON MSS: NSRDCPL,UN=CSYS

OBJECT

EDITLIB USER LIBRARY: NSRDC

SUBROUTINE 'CONTRCT'

PURPOSE

SQUEEZE ARRAY OF 1R-FORMAT CHARACTERS TO LEFT

FUNCTIONAL CATEGORIES: M4

LANGUAGE: FORTRAN IV EXTENDED

COMPUTERS (OPERATING SYSTEMS)

CDC 6000/CYBER 170 (NOS/BE)

REMARKS

SEE SUBROUTINE 'EXPAND'.

USAGE

CALL CONTRCT (A, B, NCHAR)

DESCRIPTION OF PARAMETERS

A - INPUT ARRAY WHOSE ELEMENTS EACH CONTAIN ONE
CHARACTER IN THE RIGHT-MOST 6 BITS (1R FORMAT)
B - OUTPUT ARRAY WHOSE ELEMENTS WILL EACH CONTAIN 10
CHARACTERS FROM ARRAY A (ANY LEFT-OVER BITS OF THE
LAST WORD USED IN ARRAY B WILL BE CLEARED TO 0B)
NCHAR - NUMBER OF CHARACTERS IN ARRAY A

CM REQUIRED: 44B

SUBPROGRAMS REQUIRED

PART OF LANGUAGE

MASK MOD

OTHERS

PUTCHA - INSERT CHARACTER INTO ARRAY

ARITHMETIC STATEMENT FUNCTIONS

NWORD - COMPUTE SUBSCRIPT

AUTHOR

DAVID V SOMMER - DTNSRDC CODE 1892.2

DATE WRITTEN: 04/04/77

DATE(S) REVISED

03/23/83 - RECOMPILE AT NOS/BE LEVEL 552

LOCATION OF DECKS

SOURCE

UPDATE LIBRARY ON MSS: NSRDCPL,UN=CSYS

OBJECT

EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC,.CONTRCT,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'COUPLE'

PURPOSE

LOGICALLY CONNECT (PORTIONS OF) TWO WORDS

FUNCTIONAL CATEGORIES: R1

LANGUAGE: FORTRAN IV EXTENDED

COMPUTERS (OPERATING SYSTEMS)

CDC 6000/CYBER 170 (NOS/BE)

REMARKS

NONE

USAGE

CALL COUPLE (FL, AWORD, AB, BWORD, BB, LC, IOAC)

DESCRIPTION OF PARAMETERS

FL - NUMBER OF BITS TO PROCESS
AWORD - FIRST WORD (FROM)
AB - STARTING BIT POSITION IN AWORD
BWORD - SECOND WORD (TO)
BB - STARTING BIT POSITION IN BWORD
LC - CODE FOR LOGICAL CONNECTIVE DESIRED
 0 - PUT ZEROS INTO BWORD FIELD (0)
 1 - AND THE FIELDS (M.A)
 2 - AND THE COMPLEMENT OF A TO B (M.A*)
 3 - NUMBER OF ONE IN THE LAST FIELD (M)
 4 - AND THE COMPLEMENT OF B TO A (M*.A)
 5 - SUBSTITUTE FIELD OF A INTO B (A)
 6 - EXCLUSIVE OR
 7 - OR (M+A)
 8 - AND COMPLEMENTS (A*.B*)
 9 - IDENTITY (B=A)
 10 - SUBSTITUTE COMPLEMENT OF A INTO B (A*)
 11 - OR THE COMPLEMENT OF A TO B (M+A*)
 12 - COMPLEMENT OF B (M*)
 13 - OR A TO THE COMPLEMENT OF B (A+M*)
 14 - OR THE COMPLEMENTS OF A AND B (A*+M*)
 15 - PUT ONES INTO BWORD FIELD (1)
IOAC - OUTPUT NUMBER OF ONE-BITS FOR LC=3

CM REQUIRED: 223B

SUBPROGRAMS REQUIRED

PART OF LANGUAGE

SHIFT

OTHERS

IAOC - COUNT ONE BITS IN A WORD

MASKIT - MULTIPLE-FIELD MASK GENERATOR

AUTHOR
NWL

DATE WRITTEN:

DATE(S) REVISED

03/23/83 - RECOMPILE AT NOS/BE LEVEL 552

LOCATION OF DECKS

SOURCE

UPDATE LIBRARY ON MSS: NSRDCPL,UN=CSYS

OBJECT

EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC,,COUPLE,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'DATCNV'

PURPOSE

CONVERT DATE FORMATS

FUNCTIONAL CATEGORIES: M2

LANGUAGE: FORTRAN IV EXTENDED

COMPUTERS (OPERATING SYSTEMS)
CDC 6000/CYBER 170 (NOS/BE)

REMARKS

MAY BE USED FOR ANY GREGORIAN DATE FROM OCT 15, 1582 THRU
FEB 28, 4000.

USEFUL IS DETERMINING THE ELAPSED NUMBER OF DAYS BETWEEN TWO
CALENDAR DATES.

MAY BE USED TO FIND THE DATE SO MANY DAYS FROM A GIVEN DATE.

IF THE DATE IS RETAINED IN A DATA BASE IN THE RELATIVE-DAY
FORM, IT CAN BE USED IN MANY COMPUTATIONS AND CONVERTED FOR
PRINTOUT WITHOUT THE NEED TO WORRY ABOUT LEAP YEARS AND
CHANGE OF CENTURY.

USAGE

CALL DATCNV (ITYPE, IYR, IMO, IDYMO, IDYRD, IDYYR, IDYWK)

DESCRIPTION OF PARAMETERS

ITYPE - TYPE OF CONVERSION DESIRED
1 - IN: IYR IMO IDYMO
OUT: IDYRD IDYYR IDYWK
2 - IN: IYR IDYYR
OUT: IMO IDYMO IDYRD IDYWK
3 - IN: IDYRD
OUT: IYR IMO IDYMO IDYYR IDYWK
IYR - YEAR (E.G., 1979)
IMO - MONTH (1 TO 12)
IDYMO - DAY-OF-MONTH (1 TO 31)
IDYRD - RELATIVE DAY
(RETURNS -1 IF ITYPE IS OUT OF RANGE)
IDYYR - DAY-OF-YEAR (1 TO 366)
IDYWK - DAY-OF-WEEK (0 TO 6, SUN IS 0)

CM REQUIRED: 165B

EXAMPLES

1. CONVERT JULY 11, 1979 TO THE OTHER FORMS:

CALL DATCNV (1, 1979, 7, 11, IDYRD, IDYYR, IDYWK)

RETURNS IDYRD=2444066 IDYYR=192 IDYWK=3 (WEDNESDAY)

2. CONVERT DAY 192 OF 1979 TO THE OTHER FORMS:

CALL DATCNV (2, 1979, IMO, IDYMO, IDYRD, 192, IDYWK)

RETURNS IMO=7 IDYMO=11 IDYRD=2444066 IDYWK=3

3. CONVERT RELATIVE DAY 2444066 TO OTHER FORMS:

CALL DATCNV (3, IYR, IMO, IDYMO, 2444066, IDYYR,
A IDYWK)

RETURNS IYR=1979 IMO=7 IDYMO=11 IDYYR=192 IDYWK=3

SUBPROGRAMS REQUIRED

PART OF LANGUAGE

NONE

OTHERS

JGDATE - JULIAN-GREGORIAN CONVERTER (MULTI-YEAR)

JULIAN - JULIAN-GREGORIAN CONVERTER (SINGLE YEAR)

WEKDAY - FIND DAY-OF-WEEK

AUTHOR

DAVID V SOMMER - DTNSRDC CODE 1892.2

DATE WRITTEN: 07/11/79

DATE(S) REVISED

LOCATION OF DECKS

SOURCE

UPDATE LIBRARY ON MSS: NSRDCPL,UN=CSYS

OBJECT

EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC,,DATCNV,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'DATFMT'

PURPOSE

DATE FORMAT CONVERSION

FUNCTIONAL CATEGORIES: M2

LANGUAGE: FORTRAN IV

COMPUTERS (OPERATING SYSTEMS)
CDC 6000/CYBER 170 (NOS/BE)

REMARKS

NONE

USAGE

CALL DATFMT (FMTOLD, FMTNEW, OLD, NEW)

DESCRIPTION OF PARAMETERS (ALL INTEGERS)

FMTOLD - INPUT FORMAT (OLD) -- ONE OF

- 1 - 'MM/DD/YY'
- 2 - 'MM/DD/YY'
- 3 - 'MMDDYY'
- 1 - 'YY/MM/DD'
- 2 - 'YY/MM/DD'
- 3 - 'YYMMDD'

FMTNEW - OUTPUT FORMAT (NEW)
(SAME VALUES AS FMTOLD)

OLD - DATE TO BE CONVERTED

NEW - WILL CONTAIN CONVERTED DATE

CM REQUIRED: 146B

EXAMPLE

CHANGE MMDDYY TO YY/MM/DD:

INTEGER FMTOLD, FMTNEW, OLD, NEW

DATA OLD/ "072579"/

CALL DATFMT (3, -1, OLD, NEW)

NEW WILL CONTAIN: "79/07/25".

SUBPROGRAMS REQUIRED

PART OF LANGUAGE

AND

OR

SHIFT

OTHERS

NONE

ARITHMETIC STATEMENT FUNCTIONS (CDC CYBER)

FAST R-FORMAT DECODE (RIGHT-ADJ, ZERO-FILLED)

R21FMT

R22FMT

R23FMT

R24FMT

R25FMT

R27FMT

R28FMT

AUTHOR

DAVID V SOMMER - DTNSRDC CODE 1892.2

DATE WRITTEN: B7700: 08/08/79

CDC : 02/22/80

DATE(S) REVISED

03/23/83 - RECOMPILE AT NOS/BE LEVEL 552

LOCATION OF DECKS

SOURCE

UPDATE LIBRARY ON MSS: NSRDCPL,UN=CSYS

OBJECT

EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC,,DATFMT,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'DAYONOF'

PURPOSE

PACKAGE OF SIX SUBROUTINES TO MANIPULATE THE DAYFILE
SETTINGS

FUNCTIONAL CATEGORIES: Z0

LANGUAGE: CDC 6000 CP COMPASS

COMPUTERS (OPERATING SYSTEMS)
CDC 6000/CYBER 170 (NOS/BE)

REMARKS

WHEN DAYOFF OR DAYSV/DAYSVOF IS USED, ANY MESSAGES ALREADY
IN THE DAYFILE BUFFER ARE LOST. TO INSURE THAT THEY PRINT,
USE DAYFLSH BEFORE USING DAYOFF OR DAYSV/DAYSVOF.

WHILE DAYON AND DAYOFF ARE PROVIDED, IT IS RECOMMENDED THAT
DAYSON/DAYSOFF AND DAYRS BE USED INSTEAD. THIS WILL
ELIMINATE ANY SIDE EFFECTS.

DAYONOF IS PART OF THE OPERATING SYSTEM, NOT IN LIBRARY
NSRDC. THE DOCUMENT IS HERE FOR CONVENIENCE.

USAGE

CALL DAYFLSH	FLUSH THE DAYFILE BUFFER
CALL DAYOFF	TURN THE DAYFILE BUFFER OFF
CALL DAYON	TURN THE DAYFILE ON
CALL DAYRS	RESTORE THE DAYFILE SETTING
CALL DAYSV	SAVE THE CURRENT DAYFILE SETTING AND TURN THE DAYFILE OFF (SYNONYM: DAYSVOF)
CALL DAYSVON	SAVE THE CURRENT DAYFILE SETTING AND TURN THE DAYFILE ON

CM REQUIRED: 60B

SUBPROGRAMS REQUIRED

PART OF LANGUAGE
NONE
OTHERS
NONE

AUTHORS
SYSTEMS - DTNSRDC CODE 1892.3
(ORIGINAL CODE FOR DAYSV AND DAYRS)
DAVID V SOMMER - DTNSRDC CODE 1892.2

DATE WRITTEN: UNKNOWN

DATE(S) REVISED
09/29/82 - GENERALIZE THE SUBROUTINE
- ADD CODE FOR DAYON, DAYOFF, DAYFLSH, DAYSVON

LOCATION OF DECKS
SOURCE
CODE 1892.3
OBJECT
PART OF THE OPERATING SYSTEM (SYSLIB)

ANOTHER COPY OF THIS DOCUMENT
BEGIN,DOCGET,,NSRDC,,DAYONOF,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'DISCOT'

PURPOSE

SINGLE OR DOUBLE INTERPOLATION

FUNCTIONAL CATEGORIES: E1

LANGUAGE: FORTRAN IV

COMPUTERS (OPERATING SYSTEMS)

CDC 6000/CYBER 170 (NOS/BE)

REMARKS

GIVEN A FUNCTION WITH TWO INDEPENDENT VARIABLES, X AND Z, THIS SUBROUTINE PERFORMS KX- AND KZ-ORDER INTERPOLATION TO CALCULATE THE DEPENDENT VARIABLE. ALL SINGLE-LINE FUNCTIONS ARE READ IN AS 2 SEPARATE ARRAYS AND ALL MULTI-LINE FUNCTIONS ARE READ IN AS 3 SEPARATE ARRAYS.

WHEN TABULATING DISCONTINUOUS FUNCTIONS, THERE MUST ALWAYS BE K+1 POINTS ABOVE AND BELOW THE DISCONTINUITY IN ORDER TO GET PROPER INTERPOLATION.

WHEN TABULATING ARRAYS FOR THIS SUBROUTINE, BOTH INDEPENDENT VARIABLES MUST BE IN ASCENDING ORDER.

IN SOME ENGINEERING PROGRAMS WITH MANY TABLES, IT IS QUITE DESIRABLE TO READ IN ONE ARRAY OF X'S THAT COULD BE USED FOR ALL LINES OF A MULTI-LINE FUNCTION OR DIFFERENT FORMULA. THIS NOT ONLY SAVES MUCH TIME IN PREPARING TABULAR DATA, BUT CAN ALSO SAVE MANY LOCATIONS PREVIOUSLY USED WHEN EVERY Y-COORDINATE HAD TO HAVE A CORRESPONDING X-COORDINATE. SEE EXAMPLES.

ANOTHER FEATURE IS THE POSSIBILITY OF A MULTI-LINE FUNCTION WITH NO EXTRAPOLATION ABOVE THE TOP LINE. SEE EXAMPLES.

USAGE

CALL DISCOT (X, Z, TABX, TABY, TABZ, NC, NY, NZ, Y)

DESCRIPTION OF PARAMETERS

X - X-ARGUMENT
Z - Z-ARGUMENT
(MAY BE SAME AS X ON SINGLE LINES)
TABX - ARRAY OF X'S
TABY - ARRAY OF Y'S
TABZ - ARRAY OF Z'S
NC - CONTROL WORD (+HTU)
+ IMPLIES $NX = NY/NZ$
- IMPLIES $NX = NY$
H=0 - EXTRAPOLATE WHEN $Z > Z_{MAX}$
=1 - NO EXTRAPOLATION ABOVE Z_{MAX}
T=1 TO 7 - DEGREE INTERPOLATION IN X DIRECTION
U=1 TO 7 - DEGREE INTERPOLATION IN Z DIRECTION
NY - NUMBER OF POINTS IN Y ARRAY
NZ - NUMBER OF POINTS IN Z ARRAY
Y - OUTPUT DEPENDENT VARIABLE

AD-A148 792

COMPUTER CENTER CDC LIBRARIES/NSRD (SUBPROGRAMS)(U)
DAVID W TAYLOR NAVAL SHIP RESEARCH AND DEVELOPMENT
CENTER BET. D V SOMMER ET AL. JUN 84

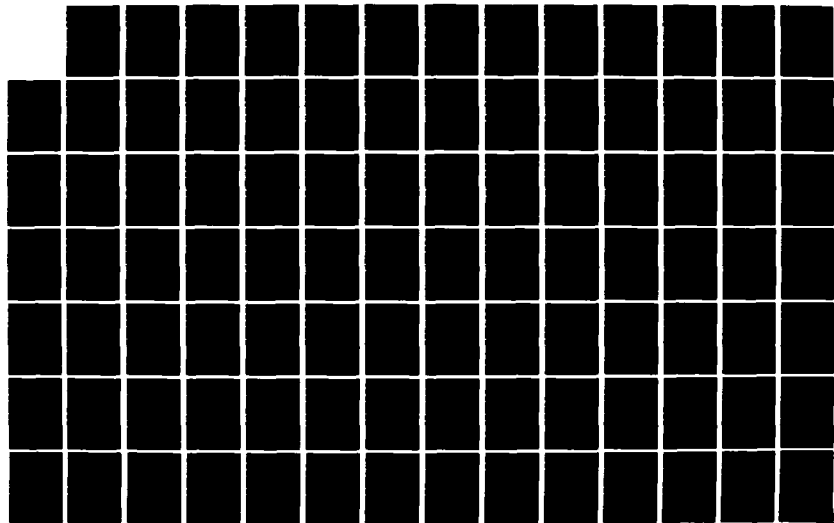
2/5

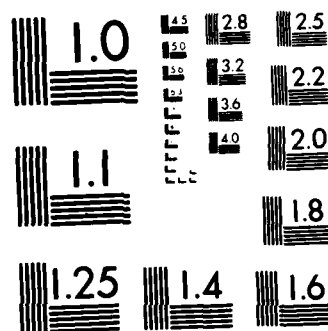
UNCLASSIFIED

DTNSRDC/CMLD-84-12

F/G 9/2

NL





MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A

CM REQUIRED: 517B

EXAMPLES

- 1) GIVEN $Y = F(X)$ KX=3
PROGRAM SAMPL1 (TAPE7,
DIMENSION TABX(50), TABY(50)
10 READ (7, 1) (TABX(I), TABY(I), I=1,50)
READ (7, 1) X
1 FORMAT (8E9.5)
CALL DISCOT (X, X, TABX, TABY, TABY, -30, 50, 0, Y)
...

2) GIVEN $Y = F(X,Z)$ KX=7, KZ=3 NX .NE. NY
PROGRAM SAMPL2 (TAPE7,
DIMENSION TABX(80), TABY(800), TABZ(10)
10 READ (7, 1) TABX
READ (7, 1) TABY
READ (7, 1) TABZ
READ (7, 1) X, Z
1 FORMAT (8E9.5)
CALL DISCOT (X, Z, TABX, TABY, TABZ, 73, 800, 10, Y)
...

3) GIVEN $Y = F(X,Z)$ KX=7, KZ=3 NX=NY
PROGRAM SAMPL3 (TAPE7,
DIMENSION TABX(800), TABY(800), TABZ(10)
10 READ (7, 1) TABX
READ (7, 1) TABY
READ (7, 1) TABZ
READ (7, 1) X, Z
1 FORMAT (8E9.5)
CALL DISCOT (X, Z, TABX, TABY, TABZ, -73, 800, 10, Y)
...

4) GIVEN $Y=F(X,Z)$ KX=7, KZ=3 NX=NY
NO EXTRAPOLATION ABOVE Z-MAX

SAME AS EXAMPLE 3 WITH 6TH PARAMETER OF CALL TO DISCOT
EQUAL TO -173.

METHOD

LAGRANGE'S INTERPOLATION FORMULA IS USED IN BOTH THE X AND
Z DIRECTION. SEE "METHODS IN NUMERICAL ANALYSIS" BY
NIELSEN.

SUBPROGRAMS REQUIRED

PART OF LANGUAGE

NONE

PART OF PROGRAM

DISSER

LAGRAN

UNS

OTHERS

NONE

AUTHOR

J. H. SUM

ALLISON DIVISION

GENERAL MOTORS CORPORATION

SHARE NUMBER 1129

DATE WRITTEN: 05/12/61

DATE(S) REVISED

09/20/83 - RECOMPILE AT NOS/BE LEVEL 552

LOCATION OF DECKS

SOURCE

UPDATE LIBRARY ON MSS: NSRDCPM,UN=CSYS (*DECK AQALL1)

OBJECT

EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC,,DISCOT,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'DMPA'

PURPOSE

CALLABLE OCTAL AND CHARACTER DUMP OF SPECIFIED PORTION
OF USER'S FIELD LENGTH (FL) (BY ACTUAL LOCATION)
(NO HEADINGS ARE PROVIDED)

FUNCTIONAL CATEGORIES: N2

LANGUAGE: FORTRAN IV

COMPUTERS (OPERATING SYSTEMS)
CDC 6000/CYBER 170 (NOS/BE)

REMARKS
NONE

USAGE

CALL DMPA (FWA, N, INIT)
CALL DMPA (FWA, N)

DESCRIPTION OF PARAMETERS

FWA - FIRST WORD ADDRESS OF AREA TO DUMP
(E.G., LOCF (ARRAY))
N - NUMBER OF WORDS TO DUMP
INIT - STARTING WORD ADDRESS TO BE PRINTED
(IF OMITTED, 0 IS USED)

CM REQUIRED: 313B

OUTPUT UNIT

UNIT #	LFN	USE
-----	-----	-----
	OUTPUT	LISTABLE OUTPUT

SUBPROGRAMS REQUIRED

PART OF LANGUAGE
LOCF

OTHERS

EQU60 - LOGICAL ARRAY COMPARE
MFETCH - READ WORK IN FL

AUTHOR

DAVID V SOMMER - DTNSRDC CODE 1892.2

DATE WRITTEN: 06/14/76

DATE(S) REVISED

03/23/83 - RECOMPILE AT NOS/BE LEVEL 552

LOCATION OF DECKS

SOURCE

UPDATE LIBRARY ON MSS: NSRDCPL,UN=CSYS

OBJECT

EDITLB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC,,DMPA,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'DMPCPA'

PURPOSE

SHORT DUMP OF JOB CONTROL POINT AREA

FUNCTIONAL CATEGORIES: N2

LANGUAGE: FORTRAN 77

COMPUTERS (OPERATING SYSTEMS)

CDC 6000/CYBER 170 (NOS/BE)

REMARKS

FOR A FULL, ANNOTATED DUMP, CALL DUMPCPA.

USAGE

CALL DMPCPA

CM REQUIRED: 324B

OUTPUT DESCRIPTION

AN OCTAL AND CHARACTER DUMP OF THE 200 (OCTAL) WORDS OF THE
CONTROL POINT AREA.

OUTPUT UNITS

UNIT #	LFN	USE
-----	-----	-----
	OUTPUT	LISTABLE OUTPUT

SUBPROGRAMS REQUIRED

PART OF LANGUAGE

DATE TIME

OTHERS

RCPA - READ CONTROL POINT AREA

AUTHOR

DAVID V SOMMER - DTNSRDC CODE 1892.2

DATE WRITTEN: 12/29/75

DATE(S) REVISED

05/25/83 - ORIGINAL SOURCE LOST - RE-WRITTEN IN FORTRAN 77

LOCATION OF DECKS

SOURCE

UPDATE LIBRARY ON MSS: NSRDCPL,UN=CSYS

OBJECT

EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC,,DMPCPA,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'DPROOT'

PURPOSE

FIND ALL ROOTS OF A REAL DOUBLE PRECISION POLYNOMIAL

FUNCTIONAL CATEGORIES: C2 B4

LANGUAGE: FORTRAN IV

COMPUTERS (OPERATING SYSTEMS)
CDC 6000/CYBER 170 (NOS/BE)

REMARKS

THE POLYNOMIAL HAS THE FORM:

$$A_1 + A_2 X + \dots + A_{N+1} X^{**N} = 0$$

USAGE

CALL DPROOT (N, A, U, V, H, B, C, CONV, NPLUS2)

DESCRIPTION OF PARAMETERS

- N - DEGREE OF THE POLYNOMIAL TO BE SOLVED
- A - DOUBLE PRECISION ARRAY (DIMENSIONED N+2) CONTAINING THE COEFFICIENTS IN THE ORDER INDICATED ABOVE
- U - DOUBLE PRECISION ARRAY (DIMENSIONED N+2) WHICH WILL CONTAIN THE REAL PARTS OF THE ROOTS
- V - DOUBLE PRECISION ARRAY (DIMENSIONED N+2) WHICH WILL CONTAIN THE IMAGINARY PARTS OF THE ROOTS
- H,B,C - DOUBLE PRECISION WORK ARRAYS (EACH DIMENSIONED N+2)
- CONV - CONVERGENCE CRITERION. INITIALLY SET BY DPROOT TO 1.0D-35 (FAR BELOW THE ACTUAL STARTING CONVERGENCE CRITERION OF 5.0D-20 (CDC 6600)). IF THE POLYNOMIAL HAS NOT CONVERGED AFTER A PRESCRIBED NUMBER OF TRIES, THE CONVERGENCE CRITERION IS RELAXED. IF, UPON EXIT FROM DPROOT, CONV IS NOT 1.0D-35, THE CONVERGENCE CRITERION HAS BEEN RELAXED TO THE NUMBER GIVEN. (CONV IS DOUBLE PRECISION.)
- NPLUS2 - MUST BE SET TO N+2

CM REQUIRED: 1031B

METHOD

THE ROUTINE CONVERGES SIMULTANEOUSLY TOWARD A LINEAR FACTOR AND A QUADRATIC FACTOR BY NEWTON'S AND BAIRSTOW'S METHODS, RESPECTIVELY. WHEN A ROOT IS FOUND BY ONE METHOD, ITERATION CONTINUES WITH BOTH METHODS USING THEIR MOST RECENT GUESSES.

SUBPROGRAMS REQUIRED

PART OF LANGUAGE

DABS DSIGN SQRT

OTHERS

NONE

AUTHOR
HARVEY ABRAMSON - NEW YORK UNIVERSITY

DATE WRITTEN: 01/66

DATE(S) REVISED
09/20/83 - RECOMPILE AT NOS/BE LEVEL 552

LOCATION OF DECKS
SOURCE
UPDATE LIBRARY ON MSS: NSRDCPM,UN=CSYS
OBJECT
EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT
BEGIN,DOCGET,,NSRDC,,DPROOT,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'DUMPA'

PURPOSE

GIVE OCTAL AND CHARACTER DUMP OF USER-SPECIFIED AREA

FUNCTIONAL CATEGORIES: N2

LANGUAGE: FORTRAN IV

COMPUTERS (OPERATING SYSTEMS)

CDC 6000/CYBER 170 (NOS/BE)

REMARKS

LINES CONTAIN 4 WORDS EACH. IF A LINE IS THE SAME AS THE PREVIOUS LINE, IT IS NOT PRINTED (UNLESS IT IS THE LAST LINE).

USAGE

CALL DUMPA (AREA, NWORDS, NAME)

DESCRIPTION OF PARAMETERS

AREA - START OF AREA TO BE DUMPED

NWORDS - NUMBER OF WORDS TO DUMP

NAME - 1-10 CHARACTER IDENTIFICATION OF START OF AREA
(E.G., 10HMYAREA(1))
(WILL BE PRINTED IN HEADING LINE)

CM REQUIRED: 240B

OUTPUT UNITS

UNIT #	LFN	USE
-----	-----	-----
	OUTPUT	LISTABLE OUTPUT

SUBPROGRAMS REQUIRED

PART OF LANGUAGE

COMPL

OTHERS

NONE

AUTHOR

DAVID V SOMMER - DTNSRDC CODE 1892.2

DATE WRITTEN: 02/06/76

DATE(S) REVISED

03/23/83 - RECOMPILE AT NOS/BE LEVEL 552

LOCATION OF DECKS

SOURCE

UPDATE LIBRARY ON MSS: NSRDCPL,UN=CSYS

OBJECT

EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC,,DUMPA,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'DUMPFL'

PURPOSE

CALLABLE OCTAL AND CHARACTER DUMP OF SPECIFIED PORTION
OF USER'S FIELD LENGTH (FL) (BY ACTUAL LOCATION)

FUNCTIONAL CATEGORIES: N2

LANGUAGE: FORTRAN IV

COMPUTERS (OPERATING SYSTEMS)
CDC 6000/CYBER 170 (NOS/BE)

REMARKS

- 1) WHEN CALLED WITHOUT AN ARGUMENT LIST, THE FTN4 CARD
FOR THE CALLING PROGRAM MUST HAVE THE 'Z' PARAMETER.
- 2) DUMP IS AT 8 LINES PER INCH ON PRINTERS WHICH WILL PRINT
AT THAT DENSITY.

USAGE

CALL DUMPFL ** SEE REMARK 1
CALL DUMPFL (LWA)
CALL DUMPFL (FWA, LWA)

DESCRIPTION OF PARAMETERS

FWA - FIRST WORD ADDRESS OF AREA TO DUMP
(SET TO ZERO IF ANY OF THE FOLLOWING:

- 1) FWA OMITTED
- 2) FWA LESS THAN ZERO
- 3) FWA GREATER THAN FL
- 4) FWA GREATER THAN LWA)

LWA - LAST WORD ADDRESS OF AREA TO DUMP
(SET TO FL IF ONE OF THE FOLLOWING:

- 1) LWA OMITTED
- 2) LWA LESS THAN OR EQUAL TO ZERO
- 3) LWA GREATER THAN FL
- 4) FWA GREATER THAN LWA)

CM REQUIRED: 376B

OUTPUT UNITS

UNIT #	LFN	USE
-----	-----	-----
	OUTPUT	LISTABLE OUTPUT

SUBPROGRAMS REQUIRED

PART OF LANGUAGE

LOCF

OTHERS

EQU60 - LOGICAL ARRAY COMPARE
FTNRFL - GET CURRENT FL
MFETCH - READ WORD IN USER'S FL

AUTHOR

DAVID V SOMMER - DTNSRDC CODE 1892.2

DATE WRITTEN: 03/12/76

DATE(S) REVISED

06/14/76

03/23/83 - RECOMPILE AT NOS/BE LEVEL 552

LOCATION OF DECKS

SOURCE

UPDATE LIBRARY ON MSS: NSRDCPL,UN=CSYS

OBJECT

EDITLB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC,,DUMPFL,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'D630I'

PURPOSE

INITIALIZE COMMON BLOCK /D630 / WITH ASCII CONTROL CODES
FOR THE DEC D630 CRT

FUNCTIONAL CATEGORIES: M4

LANGUAGE: FORTRAN 77 EXTENDED

COMPUTERS (OPERATING SYSTEMS)
CDC 6000/CYBER 170 (NOS/BE)

REMARKS

THIS SUBROUTINE MUST BE EXECUTED PRIOR TO GENERATING ASCII
MESSAGES WITH D630 CONTROL CODES USING SUBROUTINE ASCII.

COMMON BLOCK /D630 / IS OBTAINED BY RUNNING PROCEDURE ASCII0
AND INSERTING THE COMMON BLOCK INTO EACH (SUB)PROGRAM WHICH
WILL GENERATE ASCII MESSAGES HAVING D630 CONTROL CODES.

USAGE

CALL D630I

CM REQUIRED: 4B

NAMES OF D630 CONTROL CODES

MARGINS AND FORMATTING

SETTPM - SET TOP PAGE MARGIN (AT CURRENT POSITION)
SETBPM - SET BOTTOM PAGE MARGIN (AT CURRENT POSITION)
CLRBTM - CLEAR TOP AND BOTTOM PAGE MARGINS
SETLM - SET LEFT MARGIN (AT CURRENT POSITION)
SETRM - SET RIGHT MARGIN (AT CURRENT POSITION)
SETHT - SET HORIZONTAL TAB STOP (AT CURRENT POSITION)
SETVT - SET VERTICAL TAB STOP (AT CURRENT POSITION)
CLRHT - CLEAR HORIZONTAL TAB (AT CURRENT POSITION)
CLRHVT - CLEAR ALL HORIZONTAL AND VERTICAL TAB STOPS
SETLPS - SET LINES PER PAGE TO (N)
SETHMI - SET HORIZONTAL MOTION INDEX TO (N-1)
SETVMI - SET VERTICAL MOTION INDEX TO (N-1)
RSTHMI - RETURN HMI CONTROL TO SPACING SWITCH

CARRIAGE MOVEMENT

HT2N - ABSOLUTE HT TO PRINT COLUMN (N)
EAUBKP - ENABLE AUTO BACKWARD PRINTING
DAUBKP - DISABLE AUTO BACKWARD PRINTING
EAUCR - ENABLE AUTO CARRIAGE RETURN
DAUCR - DISABLE AUTO CARRIAGE RETURN
REVPRT - REVERSE PRINTING MODE
NRMPT - NORMAL PRINTING MODE
FWDPR - FORWARD PRINT MODE ON
BKWPRT - BACKWARD PRINT MODE ON/FORWARD PRINT MODE OFF
(CLEAR WITH CR)

PAPER MOVEMENT

VT2N - ABSOLUTE VT TO LINE (N)
NEGLF - PERFORM NEGATIVE LINE FEED
HALFLF - PERFORM HALF-LINE FEED
NEGLHF - PERFORM NEGATIVE HALF-LINE FEED

PRINTING

GRAFON - GRAPHICS MODE ON (CLEAR WITH CR)
GRAFOF - GRAPHICS MODE OFF
HPAMOV - HYLOT ON - ABSOLUTE MOVE (CLEAR WITH CR)
HPAPLT - HYLOT ON - ABSOLUTE PLOT (CLEAR WITH CR)
HPRMOV - HYLOT ON - RELATIVE MOVE (CLEAR WITH CR)
HPRPLT - HYLOT ON - RELATIVE PLOT (CLEAR WITH CR)
CHPLCH - CHANGE PLOT CHARACTER TO (CHARACTER)
SETPRE - SET PLOT PRECISION
COLOR2 - PRINT IN SECONDARY COLOR (RED)
COLOR1 - PRINT IN PRIMARY COLOR (BLACK)
PRTSUP - PRINT SUPPRESSION ON (CLEAR WITH CR)
LNGWHL - SELECT LANGUAGE/PRINT WHEEL SIZE

WORD PROCESSING COMMANDS

PROPON - PROPORTIONAL SPACE ON (CLEAR WITH RSTHM1)
PROPOF - PROPORTIONAL SPACE OFF
OFFSET - OFFSET SELECTION (N)
UNDRON - AUTO UNDERSCORE ON
UNDRUF - AUTO UNDERSCORE OFF
BOLD - BOLD PRINT ON (CLEAR WITH CR)
SHADOW - SHADOW PRINT ON (CLEAR WITH CR)
BOSHOF - COLD/SHADOW PRINT OFF
INCTIM - INCREASE CARRIAGE SETTling TIME TO 20MS
CLRTIM - CLEAR INCREASED CARRIAGE SETTling TIME
BK120 - BACKSPACE 1/120"
PGMODE - PROGRAM MODE ON
CANWP - CANCEL ALL WP MODES EXCEPT PROP SPACE AND CARRIA
CARRIAGE SETTling TIME
CENTR - AUTO CENTER ON (CLEAR WITH CR)
JUSTFY - AUTO JUSTIFY ON
MARGON - MARGIN CONTROL ON (OVERRIDES MARG CONT KEY)
MARGOF - MARGIN CONTROL MODE CONTROLLED BY MARG CONT KEY

MISCELLANEOUS COMMANDS

RESET - INITIATE REMOTE RESET
PRT20 - PRINT PRINT WHEEL CHARACTER HEX 20
PRT7F - PRINT PRINT WHEEL CHARACTER HEX 7F
HRISON - ENTER PROGRAM "HERE IS ..." MODE
HRISOF - EXIT PROGRAM "HERE IS ..." MODE

SUBPROGRAMS REQUIRED

PART OF LANGUAGE
NONE
OTHERS
NONE

AUTHOR

DAVID V SOMMER - DTNSRDC CODE 1892.2

DATE WRITTEN: 04/06/84

DATE(S) REVISED

LOCATION OF DECKS

SOURCE DECK

UPDATE LIBRARY ON MSS: NSRDCPL,UN=CSYS

OBJECT DECK

EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC,,D630I,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'ELLI'
SUBROUTINE 'CELLI'

PURPOSE

INCOMPLETE AND COMPLETE ELLIPTIC INTEGRALS OF THE FIRST AND
SECOND KIND

FUNCTIONAL CATEGORIES: C3

LANGUAGE: FORTRAN IV EXTENDED

COMPUTERS (OPERATING SYSTEMS)
CDC 6000/CYBER 170 (NOS/BE)

REMARKS

CELLI IS AN ENTRY POINT IN ELLI.

WHEN $ABS(PHI) \leq \pi/2$, F AND E ARE ACCURATE TO AT LEAST 10
SIGNIFICANT FIGURES. AS $ABS(PHI)$ GETS LARGE, THE ACCURACY
WILL NOT BE AS GOOD SINCE ELLI USES THE TANGENT SUBROUTINE
WHICH BECOMES LESS ACCURATE AS THE ANGLE $ABS(PHI)$ INCREASES.

USAGE

CALL ELLI (PHI, CAY, F, E)
CALL CELLI (PHI, CAY, F, E)

DESCRIPTION OF PARAMETERS

PHI - UPPER LIMIT OF INTEGRAL
(NOT USED BY CELLI WHICH ASSUMES $\pi/2$)
CAY - THE PARAMETER IN THE INTEGRAL
F - OUTPUT THE ELLIPTIC INTEGRAL OF THE FIRST KIND
(F(PHI,CAY))
E - OUTPUT THE ELLIPTIC INTEGRAL OF THE SECOND KIND
(E(PHI,CAY))

CM REQUIRED: 351B

ERROR MESSAGES

IF $K > 1$, F AND E DO NOT EXIST. A MESSAGE IS PRINTED AND
F AND E ARE SET TO PHI.

IF $K=1$ AND $ABS(PHI) \geq \pi/2$, F DOES NOT EXIST. A MESSAGE IS
PRINTED AND F IS SET TO $SIGN(PHI)*1.0E+294$. E EXISTS AND IS
COMPUTED.

OUTPUT UNITS

UNIT #	LFN	USE
-----	-----	-----
	OUTPUT	ERROR MESSAGES PRINTED BY LABRT

METHOD

WHEN $K < 1$, LANDEN'S TRANSFORMATION IS USED.

WHEN $K=1$, E IS COMPUTED BY:

$E(\text{PHI}, 1) = N + \text{ABS}(\text{SIN}(\text{PHI}) - \text{SIN}(N \cdot \text{PI}/2))$
WHERE N IS THE INTEGRAL PART OF $\text{PHI} \cdot (2/\text{PI})$.

WHEN $K=1$ AND $\text{ABS}(\text{PHI}) < \text{PI}/2$, F IS COMPUTED BY:

$$F(\text{PHI}, 1) = .5 * \text{LN} \left(\frac{1 + \text{SIN}(\text{PHI})}{1 - \text{SIN}(\text{PHI})} \right)$$

REFERENCE: "HANDBOOK OF MATHEMATICAL FUNCTIONS" BY M.
ABRAMOWITZ AND I. A. STEGUN.

SUBPROGRAMS REQUIRED

PART OF LANGUAGE

ABS	AIN	ALOG	AMIN1	AMOD
ATAN	FLOAT	INT	MOD	SIGN
SIN	SQRT	TAN		

OTHERS

LABRT - PRINT ERROR MESSAGES (57B)

AUTHORS

KARL J MELENDEZ
DUANE HARDER
LOS ALAMOS SCIENTIFIC LABORATORY

VIM ROUTINE LASL C304A

DATE WRITTEN: 02/05/68

DATE(S) REVISED

02/69 - DH
09/20/83 - RECOMPILE AT NOS/BE LEVEL 552

LOCATION OF DECKS

SOURCE

UPDATE LIBRARY ON MSS: NSRDCPM, UN=CSYS

OBJECT

EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT

BEGIN, DOCGET, , NSRDC, , ELLI, OUTPUT, MSACCES=<PASSWORD>.

SUBROUTINE 'EQPORT'

PURPOSE

GET INTERCOM EQUIPMENT AND PORT NUMBERS

FUNCTIONAL CATEGORIES: Q0

LANGUAGE: FORTRAN 77 EXTENDED

COMPUTERS (OPERATING SYSTEMS)

CDC 6000/CYBER 170 (NOS/BE)

REMARKS

THIS SUBROUTINE IS USEFUL ONLY AT DTNSRDC BECAUSE IT RELIES
ON A DTNSRDC MODIFICATION TO THE INTERCOM USER TABLE.

USAGE

CALL EQPORT (EQ, PORT)

DESCRIPTION OF PARAMETERS

EQ - INT - WILL CONTAIN THE INTERCOM EQUIPMENT NUMBER
(02 FORMAT)

PORT - INT - WILL CONTAIN THE INTERCOM PORT NUMBER
(03 FORMAT)

CM REQUIRED: 45B

SUBPROGRAMS REQUIRED

PART OF LANGUAGE
AND MASK

OTHERS

REQTBL - READ THE INTERCOM USER TABLE

ARITHMETIC STATEMENT FUNCTIONS

GETBITS - EXTRACT BITS FROM A WORD

AUTHOR

DAVID V. SOMMER - DTNSRDC CODE 1892.2

DATE WRITTEN: 10/11/84

DATE(S) REVISED

LOCATION OF DECKS

SOURCE DECK

UPDATE LIBRARY ON MSS: NSRDCPL,UN=CSYS

OBJECT DECK

EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC,,EQPORT,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'ELTIME'

PURPOSE

OBTAIN CPA, CPB, CP, PP, IO AND WALL CLOCK TIMES SINCE
LAST CALL TO ELTIME

FUNCTIONAL CATEGORIES: Q0

LANGUAGE: FORTRAN IV EXTENDED

COMPUTERS (OPERATING SYSTEMS)
CDC 6000/CYBER 170 (NOS/BE)

REMARKS
NONE

USAGE
CALL ELTIME (TIMES)

DESCRIPTION OF PARAMETER

TIMES - 7-WORD REAL ARRAY TO CONTAIN THE FOLLOWING:
1 - CPA TIME IN SECONDS
2 - CPB TIME IN SECONDS
3 - CP TIME IN SECONDS (CPA+CPB)
4 - PP TIME IN SECONDS
5 - IO TIME IN SECONDS
6 - WALL CLOCK TIME (HH.MM.SS.)
7 - WALL CLOCK TIME IN SECONDS

CM REQUIRED: 111B

SUBPROGRAMS REQUIRED

PART OF LANGUAGE
NONE

OTHERS

IHMS - CONVERT SECONDS TO ' HH.MM.SS.'
ISEC - CONVERT HH.MM.SS TO SECONDS
RCPA - READ CONTROL POINT AREA

ARITHMETIC STATEMENT FUNCTIONS

R65FMT - FAST R-FORMAT DECODE (RIGHT-ADJ, ZERO-FILLED)

AUTHOR

DAVID V SOMMER - DTNSRDC CODE 1892.2

DATE WRITTEN: 12/15/75

DATE(S) REVISED

10/31/77 - ADJUST FOR MIDNIGHT
03/02/83 - RECOMPILE AT NOS/BE LEVEL 552

LOCATION OF DECKS

SOURCE

UPDATE LIBRARY ON MSS: NSRDCPL,UN-CSYS

OBJECT

EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC,,ELTIME,OUTPUT,MSACCES=<PASSWORD>.

06/14/84

2-81

ELTIME - 1 OF 1

FUNCTION 'EQU60'

PURPOSE

LOGICAL COMPARE (OF 2 ARRAYS)

FUNCTIONAL CATEGORIES: M0

LANGUAGE: CDC 6000 CP COMPASS

COMPUTERS (OPERATING SYSTEMS)

CDC 6000/CYBER 170 (NOS/BE)

REMARKS

NONE

USAGE

TEST = EQU60 (A, B, N)

TEST = EQU60 (A, B)

DESCRIPTION OF PARAMETERS

A,B - COMPARE (ARRAY) A WITH (ARRAY) B

N - NUMBER OF WORDS TO COMPARE
(IF OMITTED, N=1)

EQU60 - WILL RETURN ONE OF:

-1. IF A .LT. B (DISPLAY CODE)

0. IF A .EQ. B (DISPLAY CODE)

+1. IF A .GT. B (DISPLAY CODE)

CM REQUIRED: 24B

SUBPROGRAMS REQUIRED

PART OF LANGUAGE

NONE

OTHERS

NONE

AUTHOR

C. FLINK - NWL - KPS

DATE WRITTEN: 12/08/70

DATE(S) REVISED

03/23/83 - RECOMPILE AT NOS/BE LEVEL 552

LOCATION OF DECKS

SOURCE

UPDATE LIBRARY ON MSS: NSRDCPL,UN=CSYS (*DECK COMPAB)

OBJECT

EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC,,EQU60,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'EXPAND'

PURPOSE

EXPAND CHARACTER STRING INTO ARRAY OF 1R-FORMAT WORDS

FUNCTIONAL CATEGORIES: M4

LANGUAGE: FORTRAN IV EXTENDED

COMPUTERS (OPERATING SYSTEMS)

CDC 6000/CYBER 170 (NOS/BE)

REMARKS

SEE SUBROUTINE 'CONTRCT'.

USAGE

CALL EXPAND (A, B, NCHAR)

DESCRIPTION OF PARAMETERS

A - INPUT ARRAY CONTAINING THE CHARACTER STRING

B - OUTPUT ARRAY WHOSE ELEMENTS WILL EACH CONTAIN ONE
CHARACTER FROM ARRAY A IN 1R FORMAT

NCHAR - NUMBER OF CHARACTERS IN ARRAY A

CM REQUIRED: 26B

SUBPROGRAMS REQUIRED

PART OF LANGUAGE

NONE

OTHERS

GETCHA - GET CHARACTER FROM ARRAY

AUTHOR

DAVID V SOMMER - DTNSRDC CODE 1892.2

DATE WRITTEN: 04/04/77

DATE(S) REVISED

03/23/83 - RECOMPILE AT NOS/BE LEVEL 552

LOCATION OF DECKS

SOURCE

UPDATE LIBRARY ON MSS: NSRDCPL,UN-CSYS

OBJECT

EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC,,EXPAND,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'EXPRM'

PURPOSE

EXTRACT PARAMETER FROM CONTROL CARD

FUNCTIONAL CATEGORIES: M4

LANGUAGE: FORTRAN IV EXTENDED

COMPUTERS (OPERATING SYSTEMS)

CDC 6000/CYBER 170 (NOS/BE)

REMARKS

ON EACH CALL, THE NEXT PARAMETER IS PASSED FROM RA+70B TO WORD(S) IAD, LEFT JUSTIFIED, ZERO-FILLED. ONCE A TERMINATOR IS ENCOUNTERED OR THE END OF A CARD IS REACHED, ZERO IS RETURNED.

IF CALLED WITH THE SECOND ARGUMENT, RETURNED IN ICC WILL BE A CODE INDICATING THE TYPE OF THE SEPARATOR FOUND FOLLOWING THE PARAMETER RETURNED IN IAD.

USAGE

CALL EXPRM (IAD)

CALL EXPRM (IAD, ICC)

DESCRIPTION OF PARAMETERS

IAD - WILL CONTAIN THE NEXT PARAMETER FROM THE CONTROL CARD. IF TERMINATOR OR END OF CARD, 0 IS RETURNED.

ICC - IF PRESENT, WILL CONTAIN A CODE INDICATING THE TYPE OF SEPARATOR ENCOUNTERED

DEC	OCT	SEPARATOR
1	1	,
2	2	=
3	3	/
4	4	(
5	5	+
6	6	-
7	7	BLANK
8	10B	;
14	16B	OTHER
15	17B	. OR) (TERMINATOR)

CM REQUIRED: 274R

SUBPROGRAMS REQUIRED

PART OF LANGUAGE

LOCF SHIFT

OTHERS

MFETCH - FETCH WORD IN USER'S FL

AUTHOR

C FLINK - KPS - NWL

DATE WRITTEN: 06/73

DATE(S) REVISED

03/23/83 - RECOMPILE AT NOS/BE LEVEL 552

LOCATION OF DECKS

SOURCE

UPDATE LIBRARY ON MSS: NSRDCPL,UN=CSYS

OBJECT

EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC,,EXPRM,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'EXTBIT'

PURPOSE

EXTRACT BITS FROM A WORD

FUNCTIONAL CATEGORIES: M4

LANGUAGE: FORTRAN IV

COMPUTERS (OPERATING SYSTEMS)

CDC 6000/CYBER 170 (NOS/BE)

REMARKS

IF NBITS GOES PAST THE END OF THE WORD, EXTBIT WILL FILL
WITH ZEROS. THERE IS NO CHECK FOR THIS.

USAGE

CALL EXTBIT (ISTART, NBITS, IN, IOUT, IRC)

DESCRIPTION OF PARAMETERS

ISTART - FIRST/ONLY BIT TO EXTRACT

(BITS ARE NUMBERED 59-0)

NBITS - NUMBER OF BITS TO EXTRACT (1-60)

IN - INPUT WORD FROM WHICH BITS ARE TO BE EXTRACTED

OUT - OUTPUT ARRAY OF DIMENSION NBITS

IRC - RETURN CODE

0 - NO ERROR

1 - ISTART OUT OF RANGE (MUST BE 0-59)

2 - NBITS OUT OF RANGE (MUST BE 1-60)

3 - BOTH ISTART AND NBITS OUT OF RANGE

CM REQUIRED: 44B

SUBPROGRAMS REQUIRED

PART OF LANGUAGE

MAX0

MIN0

MASK

SHIFT

OTHERS

NONE

AUTHOR

DAVID V SOMMER - DTNSRDC CODE 1892.2

DATE WRITTEN: 12/09/75

DATE(S) REVISED

03/23/83 - RECOMPILE AT NOS/BE LEVEL 552

LOCATION OF DECKS

SOURCE

UPDATE LIBRARY ON MSS: NSRDCPL,UN=CSYS

OBJECT

EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC,,EXTBIT,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'EXTPRM'

PURPOSE

EXTRACT NEXT PARAMETER FROM USER-SUPPLIED PARAMETER STRING

FUNCTIONAL CATEGORIES: M4

LANGUAGE: FORTRAN IV

COMPUTERS (OPERATING SYSTEMS)

CDC 6000/CYBER 170 (NOS/BE)

REMARKS

THE SUBROUTINE IS PRE-INITIALIZED FOR PROCESSING THE FIRST USER PARAMETER STRING. IF A SECOND STRING IS TO BE PROCESSED, THE SUBROUTINE MUST BE RE-INITIALIZED USING EITHER THE THIRD OR FOURTH FORM OF THE CALL.

USAGE

CALL EXTPRM (IAREA, LAREA, IPARM, ISEP)
CALL EXTPRM (IAREA, LAREA, IPARM)
CALL EXTPRM (0 , LAREA)
CALL EXTPRM (0)

DESCRIPTION OF PARAMETERS

IAREA - IN - ARRAY CONTAINING PARAMETER STRING

LAREA - IN - NUMBER OF WORDS IN 'IAREA'

OUT - FIRST AND SECOND FORMS OF CALL ONLY:

0 IF END OF 'IAREA' REACHED

THIRD FORM OF CALL:

INITIALIZE FOR THIS MANY WORDS

FOURTH FORM OF CALL (OMITTED):

INITIALIZE FOR 16 WORDS

(BECAUSE 'LAREA' IS BOTH AN INPUT AND OUTPUT ARGUMENT, IT MUST ALWAYS BE USED AS A VARIABLE, NEVER AS AN EXPLICIT INTEGER.)

IPARM - OUT - NEXT PARAMETER, LEFT-JUSTIFIED, ZERO-FILLED

ISEP - OUT - IF PRESENT, CODE INDICATING TYPE OF SEPARATOR FOUND FOLLOWING THE PARAMETER RETURNED IN 'IPARM' (COMPATIBLE WITH SCOPE 3.3 AND 3.4)

DEC	OCT	SEPARATOR
1	01	,
2	02	=
3	03	/
4	04	(
5	05	+
6	06	-
7	07	BLANK
8	10	;
14	16	OTHER
15	17	. OR) (TERMINATOR)

CM REQUIRED: 425B

SUBPROGRAMS REQUIRED

PART OF LANGUAGE

LOCF MINO SHIFT

OTHERS

NONE

AUTHORS

C FLINK - KPS NWL

D V SOMMER - DTNSRDC CODE 1892.2

DATE WRITTEN: 06/73 - CF

DATE(S) REVISED

04/11/74 - DVS - ORIGINAL SUBROUTINE 'EXPRM' MODIFIED TO
ACCEPT USER-SUPPLIED PARAMETER STRING

03/21/83 - RECOMPILE AT NOS/BE LEVEL 552

LOCATION OF DECKS

SOURCE

UPDATE LIBRARY ON MSS: NSRDCPL,UN=CSYS

OBJECT

EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC,,EXTPRM,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'FBINRD'

PURPOSE

UNPACK AN INPUT ARRAY

FUNCTIONAL CATEGORIES: M4

LANGUAGE: FORTRAN IV EXTENDED

COMPUTERS (OPERATING SYSTEMS)

CDC 6000/CYBER 170 (NOS/BE)

REMARKS

NONE

USAGE

CALL FBINRD (BW, NUMB, IN, OUT)

DESCRIPTION OF PARAMETERS

BW - BITS-PER-WORD TO BE EXTRACTED

NUMB - NUMBER OF BW-BIT OUTPUT WORDS DESIRED

DIMENSION OF IN IS $((NUMB * BW) + 59) / 60$

DIMENSION OF OUT IS NUMB

IN - INPUT ARRAY

OUT - OUTPUT ARRAY

CM REQUIRED: 35B

METHOD

THE BW EXTRACTED BITS ARE RIGHT JUSTIFIED WITH LEADING
ZEROS IN OUT.

SUBPROGRAMS REQUIRED

PART OF LANGUAGE

NONE

OTHERS

NONE

AUTHOR

A. CINCOTTA - DTNSRDC CODE 1892.3

DATE WRITTEN: 03/75

DATE(S) REVISED

LOCATION OF DECKS

SOURCE

UPDATE LIBRARY ON MSS: NSRDCPL, UN=CSYS

OBJECT

EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT

BEGIN, DOCGET, , NSRDC, , FBINRD, OUTPUT, MSACCES=<PASSWORD>.

SUBROUTINE 'FFT'

PURPOSE

FAST FOURIER TRANSFORM FOR COMPLEX TABULATED FUNCTION

FUNCTIONAL CATEGORIES: E2

LANGUAGE: FORTRAN IV EXTENDED

COMPUTERS (OPERATING SYSTEMS)

CDC 6000/CYBER 170 (NOS/BE)

REMARKS

THIS ROUTINE ALSO COMPUTES THE INVERSE FOURIER TRANSFORM.
WITH SLIGHT MODIFICATIONS OF THE RESULTING TRANSFORM, TWO
REAL TABULATED FUNCTIONS MAY BE TRANSFORMED SIMULTANEOUSLY.

FOR REAL, ONE-DIMENSIONAL DATA, SEE RFFT OR RFSN.

USAGE

CALL FFT (A, M, INV, S, IFSET, IFERR)

DESCRIPTION OF PARAMETERS

A - THE ARRAY CONTAINING A COMPLEX TABULATED FUNCTION OF
UP TO 3 DIMENSIONS TO BE TRANSFORMED. 'A' CONTAINS
CONSECUTIVE COMPLEX PAIRS OF DATA. FOR THE ARRAY
A(I,J,K), THE ELEMENT WITH SUBSCRIPT (I,J,K) IS
STORED WITH THE REAL PART IN SUBSCRIPT $2*((K*N1*N2)+$
 $(J*N1) + I) + 1$, AND THE IMAGINARY PART IN THE
FOLLOWING CELL. N1 AND N2 ARE COMPUTED AS $2**M(1)$
AND $2**M(2)$, RESPECTIVELY. NOTE THAT 'I' VARIES
MOST RAPIDLY, K LEAST RAPIDLY.

ON OUTPUT, 'A' CONTAINS THE FOURIER TRANSFORM.

M - A 3-CELL ARRAY WHICH CONTAINS THE MINIMUM INTEGER
WHICH IS GE THE LOG-BASE-2 OF THE DIMENSIONS OF 'A'.

INV - SCRATCH ARRAY REQUIRING 1/8 THE DIMENSION OF 'A'

S - SCRATCH ARRAY REQUIRING 1/8 THE DIMENSION OF 'A'

IFSET - COMPUTATION FLAG

= 0 -- SET UP TABLES IN INV AND S

= 1 -- SET UP TABLES AND COMPUTE FOURIER TRANSFORM

=-1 -- SET UP TABLES AND COMPUTE INVERSE FOURIER
TRANSFORM

= 2 -- COMPUTE FOURIER TRANSFORM ASSUMING TABLES
EXIST

=-2 -- COMPUTE INVERSE FOURIER TRANSFORM ASSUMING
TABLES EXIST

IFERR - RETURN CODE

= 0 -- NORMAL COMPLETION

<>0 -- ERRORS IN SUBROUTINE ARGUMENTS

NOTE: $3 \leq M(L) \leq 20$, WHERE L IS THE SUBSCRIPT OF THE
LARGEST ELEMENT IN M. DATA DIMENSIONS MUST BE POWERS
OF 2. IF DATA DIMENSIONS ARE $< 2**M(L)$, THE
REMAINING LOCATIONS MUST BE SET TO ZERO OR ANY
APPROPRIATE CONSTANT.

CM REQUIRED: 1246B

METHOD

THIS SUBROUTINE IS BASED ON AN ALGORITHM PROPOSED BY COOLEY AND TUKEY AND IS WELL DOCUMENTED IN REFERENCE 1. BASICALLY, THE ALGORITHM DECOMPOSES THE TRANSFORMATION INTO PRODUCT OF SEVERAL ELEMENTARY TRANSFORMATIONS FOLLOWED BY A REORDERING OF SUBSCRIPTS OF THE RESULT.

A METHOD EXISTS FOR TRANSFORMING 2 REAL DATA SETS SIMULTANEOUSLY WITH AN ELEMENTARY TRANSFORMATION ON THE RESULTING ANSWERS TO SEPARATE THE TRANSFORMS. THIS PROCEDURE IS DOCUMENTED IN REFERENCE 2.

TWO OTHER ROUTINES RFFT AND RFSN ACCOMPLISH THE FAST FOURIER TRANSFORM AND INVERSE TRANSFORM, RESPECTIVELY, OF ONE-DIMENSIONAL DATA. THESE ROUTINES USE A MODIFICATION OF THE COOLEY-TUKEY PROCESS AND ARE FASTER THAN PROCESSING A COMPLEX ARRAY WITH A ZERO IMAGINARY COMPONENT.

REFERENCES

1. COOLEY, J. W., AND TUKEY, J. W., "AN ALGORITHM FOR THE MACHINE CALCULATION OF COMPLEX FOURIER SERIES," MATH. COMPUT. 19, 90 (APRIL 1965), 297-301.
2. SINGLETON, RICHARD C., "ON COMPUTING THE FAST FOURIER TRANSFORM," COMM. OF THE ACM, VOL, 10, NO. 10, OCTOBER 1967.
3. SYSTEM/360 SCIENTIFIC SUBROUTINE PACKAGE, IBM TECHNICAL PUBLICATIONS DEPARTMENT, 1967.

SUBPROGRAMS REQUIRED

PART OF LANGUAGE

COS IABS MAX0 SIN SQRT

OTHERS

NONE

AUTHORS

WES RICE
DUANE HARDER
LOS ALAMOS SCIENTIFIC LABORATORY

VIM ROUTINE LASL C329A

DATE WRITTEN: 07/16/68

DATE(S) REVISED

02/69 - DH

09/20/83 - RECOMPILE AT NOS/BE LEVEL 552

LOCATION OF DECKS

SOURCE

UPDATE LIBRARY ON MSS: NSRDCPM,UN=CSYS (*DECK LASC329)

OBJECT

EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC,,FFT,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'FFT5'

PURPOSE

FAST FOURIER TRANSFORM

FUNCTIONAL CATEGORIES: E2

LANGUAGE: FORTRAN IV EXTENDED

COMPUTERS (OPERATING SYSTEMS)
CDC 6000/CYBER 170 (NOS/BE)

REMARKS

NONE

USAGE

CALL FFT5 (F, NPTS, KOMPLX)

DESCRIPTION OF PARAMETERS

F - (COMPLEX) ARRAY TO BE TRANSFORMED
(IF 'F' IS REAL, THE VALUES MUST BE STORED IN
CONTIGUOUS CORE LOCATIONS)
NPTS - NUMBER OF WORDS IN 'F' TO BE TRANSFORMED.
MUST BE POWER OF 2 AND LE 8192.
TO COMPUTE THE INVERSE TRANSFORM, NPTS MUST
BE NEGATIVE.
KOMPLX - ONE OF:
0 - DATA IN 'F' IS REAL
1 - DATA IN 'F' IS COMPLEX

CM REQUIRED: 4520B

METHOD

SEE CMD-25-71

SUBPROGRAMS REQUIRED

PART OF LANGUAGE
COS FLOAT IABS
PART OF PROGRAM
IRVING
OTHERS
NONE

AUTHORS

W. H. HAILE
GEORGE GLUCK

DATE WRITTEN: 1971

DATE(S) REVISED

09/30/83 - RECOMPILE AT NOS/BE LEVEL 552

LOCATION OF DECKS

SOURCE

UPDATE LIBRARY ON MSS: NSRDCPM,UN=CSYS (*DECK AMFFT5)

OBJECT

EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC,,FFT5,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'FINDC'

PURPOSE

FIND PRESENCE OR ABSENCE OF SPECIFIED CHARACTER IN AN ARRAY
(USER SPECIFIES RELATIONAL OPERAND)

FUNCTIONAL CATEGORIES: M5

LANGUAGE: FORTRAN IV

COMPUTERS (OPERATING SYSTEMS)
CDC 6000/CYBER 170 (NOS/BE)

REMARKS
NONE

USAGE

CALL FINDC (A, NA, CHAR, NC, NW, REL, FIRSTCH)
CALL FINDC (A, NA, CHAR, NC, NW, REL)

DESCRIPTION OF PARAMETERS

A - ARRAY TO BE SEARCHED
NA - NUMBER OF WORDS IN 'A' TO BE SEARCHED
CHAR - CHARACTER TO BE SEARCHED FOR ACCORDING TO 'REL'
(LEFT-ADJ, BLANK- OR ZERO-FILLED -OR-
RIGHT-ADJ, ZERO-FILLED)
NC - OUTPUT POSITION OF FIRST CHARACTER (RELATIVE TO
START OF 'A') WHICH SATISFIES THE RELATION
'REL' -OR-
0 - CONDITION IS NOT SATISFIED -OR-
-1 - 'REL' IS INVALID
-2 - 'FIRSTCH' GT 10*NA
NW - OUTPUT SUBSCRIPT OF WORD CONTAINING POSITION
'NC' -OR-
0 - CONDITION IS NOT SATISFIED -OR-
-1 - 'REL' IS INVALID
-2 - 'FIRSTCH' GT 10*NA
REL - RELATIONAL OPERAND
"EQ" - FIND FIRST CHARACTER IN 'A' EQUAL TO
'CHAR'
"NE" - FIND FIRST CHARACTER IN 'A' NOT EQUAL TO
'CHAR'
"LT" - FIND FIRST CHARACTER IN 'A' LESS THAN
'CHAR'
"LE" - FIND FIRST CHARACTER IN 'A' LESS THAN OR
EQUAL TO 'CHAR'
"GT" - FIND FIRST CHARACTER IN 'A' GREATER THAN
'CHAR'
"GE" - FIND FIRST CHARACTER IN 'A' GREATER THAN
OR EQUAL TO 'CHAR'
FIRSTCH - FIRST CHARACTER TO BE SEARCHED (OPTIONAL)
(DEFAULT: 1)
IF FIRSTCH < 1, DEFAULT IS USED.

CM REQUIRED: 236B

SUBPROGRAMS REQUIRED

PART OF LANGUAGE

LOCF

OTHERS

GETCHA - GET CHARACTER FROM ARRAY

ARITHMETIC STATEMENT FUNCTIONS

L11FMT - FAST L-FORMAT DECODE (LEFT-ADJ, ZERO-FILLED)

L21FMT - FAST L-FORMAT DECODE (LEFT-ADJ, ZERO-FILLED)

R11FMT - FAST R-FORMAT DECODE (RIGHT-ADJ, ZERO-FILLED)

AUTHORS

DAVID V SOMMER - DTNSRDC CODE 1892.2

PETE ROTH - DTNSRDC CODE 1720.3

DATE WRITTEN: 04/20/76

DATE(S) REVISED

07/22/76 - PR/DVS - ADD PARAMETER 'FIRSTCH'

03/23/83 - RECOMPILE AT NOS/BE LEVEL 552

LOCATION OF DECKS

SOURCE

UPDATE LIBRARY ON MSS: NSRDCPL,UN=CSYS

OBJECT

EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC,,FINDC,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'FINDW'

PURPOSE

FIND PRESENCE OR ABSENCE OF SPECIFIED WORD IN AN ARRAY
(USER SPECIFIES RELATIONAL OPERAND)

FUNCTIONAL CATEGORIES: M5

LANGUAGE: FORTRAN IV

COMPUTERS (OPERATING SYSTEMS)
CDC 6000/CYBER 170 (NOS/BE)

REMARKS
NONE

USAGE

CALL FINDW (A, NA, W, NW, REL)

DESCRIPTION OF PARAMETERS

A - ARRAY TO BE SEARCHED
NA - NUMBER OF WORDS IN 'A' TO BE SEARCHED
W - WORD TO BE TESTED FOR ACCORDING TO 'REL'
NW - OUTPUT POSITION (SUBSCRIPT) OF FIRST WORD IN 'A'
WHICH SATISFIES THE RELATION 'REL' -OR-
0 - CONDITION IS NOT SATISFIED -OR-
-1 - 'REL' IS INVALID
REL - RELATIONAL OPERAND
"EQ" - FIND FIRST WORD IN 'A' WHICH IS EQUAL TO 'W'
"NE" - FIND FIRST WORD IN 'A' WHICH IS NOT EQUAL TO
'W'
"LT" - FIND FIRST WORD IN 'A' WHICH IS LESS THAN 'W'
"LE" - FIND FIRST WORD IN 'A' WHICH IS LESS THAN OR
EQUAL TO 'W'
"GT" - FIND FIRST WORD IN 'A' WHICH IS GREATER THAN
'W'
"GE" - FIND FIRST WORD IN 'A' WHICH IS GREATER THAN
OR EQUAL TO 'W'

CM REQUIRED: 142B

SUBPROGRAMS REQUIRED

PART OF LANGUAGE

NONE

OTHERS

NONE

AUTHOR

DAVID V SOMMER - DTNSRDC CODE 1892.2

DATE WRITTEN: 02/20/76

DATE(S) REVISED

03/23/83 - RECOMPILE AT NOS/BE LEVEL 552

LOCATION OF DECKS

SOURCE

UPDATE LIBRARY ON MSS: NSRDCPL,UN=CSYS

OBJECT

EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC,,FINDW,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'FINDWRD'

PURPOSE

FIND SPECIFIED WORD IN AN ARRAY

FUNCTIONAL CATEGORIES: M5

LANGUAGE: FORTRAN IV

COMPUTERS (OPERATING SYSTEMS)

CDC 6000/CYBER 170 (NOS/BE)

REMARKS

NONE

USAGE

CALL FINDWRD (A, NA, WORD, NWORD)

CALL FINDWR (A, NA, WORD, NWORD)

DESCRIPTION OF PARAMETERS

A - ARRAY TO BE SEARCHED

NA - NUMBER OF WORDS IN 'A' TO BE SEARCHED

WORD - WORD TO BE SEARCHED FOR

NWORD - OUTPUT SUBSCRIPT OF FIRST OCCURRENCE OF WORD
IN 'A' (IF NO MATCH, ZERO (0) IS RETURNED)

CM REQUIRED: 22B

SUBPROGRAMS REQUIRED

PART OF LANGUAGE

NONE

OTHERS

NONE

AUTHOR

DAVID V SOMMER - DTNSRDC CODE 1892.2

DATE WRITTEN: 07/08/74

DATE(S) REVISED

05/07/79 - MOVE TO BURROUGHS B7700

03/21/83 - RECOMPILE AT NOS/BE LEVEL 552

LOCATION OF DECKS

SOURCE

UPDATE LIBRARY ON MSS: NSRDCPL,UN=CSYS

OBJECT

EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC,,FINDWRD,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'FRESNEL'

PURPOSE

EVALUATE FRESNEL INTEGRALS

FUNCTIONAL CATEGORIES: C3

LANGUAGE: FORTRAN IV

COMPUTERS (OPERATING SYSTEMS)

CDC 6000/CYBER 170 (NOS/BE)

REMARKS

$C(X) = \text{INTEGRAL (FROM 0 TO X) } \cos((\pi/2)U^{**2})DU$

$S(X) = \text{INTEGRAL (FROM 0 TO X) } \sin((\pi/2)U^{**2})DU$

RELATIVE ERROR < 2.E-10.

USAGE

CALL FRESNEL (X, C, S)

DESCRIPTION OF PARAMETERS

X - REAL INPUT PARAMETER

C - REAL OUTPUT PARAMETER (C(X))

S - REAL OUTPUT PARAMETER (S(X))

CM REQUIRED: 252B

METHOD

TRUNCATED CHEBYSHEV SERIES

REFERENCE

BULRISCH, R., "NUMERICAL CALCULATION OF THE SINE, COSINE AND FRESNEL INTEGRALS", NUMERISCHE MATHEMATIK, 9, 1967, PP. 380-385.

SUBPROGRAMS REQUIRED

PART OF LANGUAGE

ABS AINT COS FLOAT SIN
OTHERS
NONE

AUTHOR

R BULIRSCH - UNIVERSITY OF CALIFORNIA AT SAN DIEGO

DATE WRITTEN: 01/68

DATE(S) REVISED

09/20/83 - RECOMPILE AT NOS/BE LEVEL 552

LOCATION OF DECKS

SOURCE

UPDATE LIBRARY ON MSS: NSRDCPM,UN=CSYS

OBJECT

EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC,,FRESNEL,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'FTNRFL'

PURPOSE

GET/SET CORE SIZE

FUNCTIONAL CATEGORIES: Q0

LANGUAGE: CDC 6000 CP COMPASS

COMPUTERS (OPERATING SYSTEMS)

CDC 6000/CYBER 170 (NOS/BE)

REMARKS

NONE

USAGE

CALL FTNRFL (IFL)

DESCRIPTION OF PARAMETER

IFL - INTEGER FIELD LENGTH DESIRED.

IF THE VALUE OF IFL IS ZERO (0), THE FL IS NOT CHANGED
BUT THE PRESENT FIELD LENGTH IS RETURNED IN IFL.

CM REQUIRED: 20B

SUBPROGRAMS REQUIRED

PART OF LANGUAGE

NONE

OTHERS

NONE

AUTHOR

C FLINK - KPS NWL

DATE WRITTEN: 12/18/70

DATE(S) REVISED

03/23/83 - RECOMPILE AT NOS/BE LEVEL 552

LOCATION OF DECKS

SOURCE

UPDATE LIBRARY ON MSS: NSRDCPL,UN=CSYS

OBJECT

EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC,,FTNRFL,OUTPUT,MSACCES=<PASSWORD>.

FUNCTION 'GAMCAR'

PURPOSE

COMPLEX GAMMA FUNCTION OF A COMPLEX ARGUMENT HAVING POSITIVE
REAL PART

FUNCTIONAL CATEGORIES: C3

LANGUAGE: FORTRAN IV

COMPUTERS (OPERATING SYSTEMS)
CDC 6000/CYBER 170 (NOS/BE)

REMARKS

HAS BEEN CHECKED FOR $CX = A + BI$, $0 < A \leq 20$, $0 < B \leq 20$.
RELATIVE ERROR IS $\leq 2 \times 10^{-10}$.

USAGE

COMPLEX CX, CY, GAMCAR

...
CY = GAMCAR (CX)

DESCRIPTION OF PARAMETERS

CX - COMPLEX VARIABLE WITH POSITIVE REAL PART
CY - COMPLEX SOLUTION

CM REQUIRED: 154B

METHOD

$$\text{GAMCAR}(Z+1) = (Z+5.5)^{(Z+1/2)} * \\ E^{*- (Z+5.5)} * \\ \text{SQRT}(2*PI) * \\ (\text{CONSTANT} + \text{SUM } (I=1,6) (CI/Z+I))$$

WHERE CONSTANT = 1.00000 00001 78
C(1) = 76.18009 17294 06
C(2) = -86.50532 03271 12
C(3) = 24.01409 82222 3
C(4) = -1.23173 95161 4
C(5) = 0.00120 85800 3
C(6) = -0.00000 53638 2

REFERENCES

C. LANCZOS, NUMERICAL ANALYSIS, SIAM SERIES B, VOL 1, PP.
86-96, 1964.

HANDBOOK OF MATHEMATICAL FUNCTIONS, NATIONAL BUREAU OF
STANDARDS, APPLIED MATHEMATICS SERIES NO. 55.

SUBPROGRAMS REQUIRED

PART OF LANGUAGE
CEXP CLOG
OTHERS
NONE

AUTHORS

R L PEXTON - LAWRENCE RADIATION LABORATORY
D A WILBER - LAWRENCE RADIATION LABORATORY

DATE WRITTEN: 12/16/64 (RLP)

DATE(S) REVISED

08/65 (DAW)
09/20/83 - RECOMPILE AT NOS/BE LEVEL 552

LOCATION OF DECKS

SOURCE

UPDATE LIBRARY ON MSS: NSRDCPM,UN=CSYS

OBJECT

EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC,,GAMCAR,OUTPUT,MSACCES=<PASSWORD>.

FUNCTION 'GAMMA'

PURPOSE

INCOMPLETE GAMMA FUNCTION

FUNCTIONAL CATEGORIES: C3

LANGUAGE: FORTRAN IV

COMPUTERS (OPERATING SYSTEMS)
CDC 6000/CYBER 170 (NOS/BE)

REMARKS

COMPUTES GAMMA (A, X) UNDER THE FOLLOWING RESTRICTIONS:

- 1) $X \geq 0$,
- 2) WHEN $X = 0$, A IS NOT A NON-POSITIVE INTEGER.

USAGE

Y = GAMMA (A, X)

DESCRIPTION OF PARAMETERS

A - FLOATING POINT NUMBER
X - ≥ 0 (X=0 FOR COMPLETE GAMMA FUNCTION)

CM REQUIRED: 520B

OUTPUT UNIT

UNIT #	LFN	USE
-----	-----	-----
	OUTPUT	ERROR MESSAGES

REFERENCE

C. E. FRÜBERG, RATIONAL CHEBYCHEV APPROXIMATION OF
ELEMENTARY FUNCTIONS, BIT, VOL. 1, P. 256, 1961.

SUBPROGRAMS REQUIRED

PART OF LANGUAGE

ABS ALOG SQRT

PART OF PROGRAM

GAMNEG - COMPUTES GAMMA(A,X) WHEN A IS NEGATIVE INTEGER
(DUE TO THE REPRESENTATION OF NUMBERS IN THE
CDC, IF $A = -N + \epsilon$, WHERE $\epsilon < 1.0E-10$, THEN A IS
TAKEN TO BE A NEGATIVE INTEGER)
GCHEB - COMPUTES BY A RATIONAL CHEBYSHEV APPROXIMATION
(GAMMA(A))
GFRAC - COMPUTES THE CONTINUES FUNCTION FOR GAMMA(A,X)
GSERIES - COMPUTES SUM $(N=0, \text{INF}) ((-X)^N / ((A+N)N!))$

OTHERS

NONE

AUTHOR

HARVEY ABRAMSON - NEW YORK UNIVERSITY

DATE WRITTEN: 05/15/66

DATE(S) REVISED

05/67

09/20/83 - RECOMPILE AT NOS/BE LEVEL 552

LOCATION OF DECKS

SOURCE

UPDATE LIBRARY ON MSS: NSRDCPM,UN=CSYS

OBJECT

EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC,,GAMMA,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'GAUSS'

PURPOSE

GAUSSIAN ELIMINATION WITH PARTIAL PIVOTING FOR SOLVING
AX=B WHERE B MAY BE A SYSTEM OF M RIGHT-HAND SIDES

FUNCTIONAL CATEGORIES: F4 F3

LANGUAGE: FORTRAN IV

COMPUTERS (OPERATING SYSTEMS)
CDC 6000/CYBER 170 (NOS/BE)

REMARKS

IF A-INVERSE IS DESIRED; ANX=[B IN] WILL YIELD
THE SOLUTION TO AX=B AS WELL AS THE INVERSE.

IF MM=0, XX CONTAINS RESULT OF FIRST GAUSSIAN ELIMINATION.

USAGE

CALL GAUSS (N, M, AA, BB, XX, VAL2, DET, MM)

DESCRIPTION OF PARAMETERS

N - SIZE OF MATRIX AA
M - NUMBER OF COLUMNS IN BB (≤ 51)
(NUMBER OF RIGHT HAND SIDES)
AA - MATRIX (51X51)
BB - RIGHT HAND SIDE(S) (51X51)
XX - SOLUTION VECTORS (51X51)
VAL2 - FINAL MAXIMUM ROW SUM OF RESIDUALS
(INFINITY-NORM OF RESIDUAL)
DET - DETERMINANT
MM - NUMBER OF ITERATIONS ON RESIDUALS
INPUT - MAXIMUM NUMBER TO BE PERMITTED
OUTPUT - NUMBER ACTUALLY DONE

CM REQUIRED: 17741B

REFERENCE:

WILKINSON, J. H., ROUNDING ERRORS IN ALGEBRAIC PROCESSES.

SUBPROGRAMS REQUIRED

PART OF LANGUAGE
ABS
OTHERS
NONE

AUTHORS

ROBERT MARGOLIS - UNIVERSITY OF MARYLAND
SUSAN VOIGHT - DTNSRDC

DATE WRITTEN: 1971

DATE(S) REVISED

09/20/83 - RECOMPILE AT NOS/BE LEVEL 552

LOCATION OF DECKS

SOURCE

UPDATE LIBRARY ON MSS: NSRDCPM,UN=CSYS (*DECK AMGAU2)

OBJECT

EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC,,GAUSS,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'GETCCL'

PURPOSE

GET CCL FIELDS (REGISTERS AND FLAGS)

FUNCTIONAL CATEGORIES: Q0 NO L3

LANGUAGE: FORTRAN 77 EXTENDED

COMPUTERS (OPERATING SYSTEMS)

CDC 6000/CYBER 170 (NOS/BE)

REMARKS

USES LABELLED COMMON BLOCK /ZZZCCL/.

USAGE

CALL GETCCL (FIELDS)

DESCRIPTION OF PARAMETER

FIELDS - 8-WORD INTEGER ARRAY WHICH WILL CONTAIN:

FIELDS(1) - EF (ERROR FLAG)

FIELDS(2) - EFG (GLOBAL ERROR FLAG)

FIELDS(3) - R1 (REGISTER)

FIELDS(4) - R2 (REGISTER)

FIELDS(5) - R3 (REGISTER)

FIELDS(6) - R1G (GLOBAL REGISTER)

FIELDS(7) - DSC (DAYFILE SKIP FLAG)

FIELDS(8) - PNL (PROCEDURE NESTING LEVEL)

CM REQUIRED: 45B

SUBPROGRAMS REQUIRED

PART OF LANGUAGE

AND SHIFT

OTHERS

RCPA - READ CONTROL POINT AREA

ARITHMETIC STATEMENT FUNCTIONS

FAST L-FORMAT DECODE (LEFT-ADJ, ZERO-FILLED)

L32FMT L35FMT L38FMT

FAST R-FORMAT DECODE (RIGHT-ADJ, ZERO-FILLED)

R11FMT R110FMT F25FMT

OTHERS

GETBIT - EXTRACT A BIT FROM A WORD

AUTHOR

DAVID V SOMMER - DTNSRDC CODE 1892.2

DATE WRITTEN: 04/09/81

DATE(S) REVISED

LOCATION OF DECKS

SOURCE

UPDATE LIBRARY ON MSS: NSRDCPL,UN=CSYS

OBJECT

EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC,,GETCCL,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'GETCHA'
FUNCTION 'GETCHA'

PURPOSE

EXTRACT CHARACTER FROM SPECIFIED POSITION IN AN ARRAY

FUNCTIONAL CATEGORIES: M4 M5

LANGUAGE: FORTRAN IV EXTENDED

COMPUTERS (OPERATING SYSTEMS)
CDC 6000/CYBER 170 (NOS/BE)

USAGE

CALL GETCHA (ARRAY, NPOS, ICHAR)
VARIABLE = GETCHA (ARRAY, NPOS, ICHAR)

DESCRIPTION OF PARAMETERS

ARRAY - ARRAY FROM WHICH CHARACTER IS TO BE EXTRACTED
NPOS - POSITION OF CHARACTER TO BE EXTRACTED
(POSITION 1 IS LEFT-MOST 6-BIT CHARACTER IN
ARRAY(1))
ICHAR - WILL CONTAIN THE EXTRACTED CHARACTER IN 1R FORMAT
(RIGHT-ADJ, ZERO-FILLED)
GETCHA - WHEN USED AS A FUNCTION, GETCHA WILL CONTAIN THE
SAME AS ICHAR AND MUST BE DECLARED INTEGER IN THE
CALLING PROGRAM

CM REQUIRED: 56B (INCLUDES PUTCHA)

REMARKS
NONE

SUBPROGRAMS REQUIRED
PART OF LANGUAGE
MOD SHIFT
OTHERS
NONE

AUTHOR
DAVID V SOMMER - DTNSRDC CODE 1892.2

DATE WRITTEN: 03/16/76

DATE(S) REVISED
08/01/79 - DOCUMENT MODIFIED
03/23/83 - RECOMPILE AT NOS/BE LEVEL 552

LOCATION OF DECKS
SOURCE
UPDATE LIBRARY ON MSS: NSRDCPL,UN=CSYS
OBJECT
EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT
BEGIN,DOCGET,,NSRDC,,GETCHA,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'GETCHR'
FUNCTION 'GETCHR'

PURPOSE

EXTRACT CHARACTER FROM SPECIFIED POSITION IN A WORD

FUNCTIONAL CATEGORIES: M4 M5

LANGUAGE: FORTRAN IV EXTENDED

COMPUTERS (OPERATING SYSTEMS)
CDC 6000/CYBER 170 (NOS/BE)

USAGE

CALL GETCHR (WORD, NPOS, ICHAR)
VARIABLE = GETCHR (WORD, NPOS, ICHAR)

DESCRIPTION OF PARAMETERS

WORD - WORD FROM WHICH CHARACTER IS TO BE EXTRACTED
NPOS - POSITION OF CHARACTER TO BE EXTRACTED
(POSITION 1 IS LEFT-MOST 6-BIT CHARACTER IN WORD)
ICHAR - WILL CONTAIN THE EXTRACTED CHARACTER IN 1H FORMAT
(LEFT-ADJ, BLANK-FILLED)
GETCHR - WHEN USED AS A FUNCTION, GETCHR WILL CONTAIN THE
SAME AS ICHAR AND MUST BE DECLARED INTEGER IN THE
CALLING PROGRAM

CM REQUIRED: 46B (INCLUDES PUTCHR)

REMARKS

NONE

SUBPROGRAMS REQUIRED

PART OF LANGUAGE
SHIFT
OTHERS
NONE

AUTHOR

FROM BIMED PACKAGE

DATE WRITTEN:

1975 - DAVID V SOMMER - DTNSRDC CODE 1892.2

DATE(S) REVISED

08/01/79 - DOCUMENT MODIFIED
03/23/83 - RECOMPILE AT NOS/BE LEVEL 552

LOCATION OF DECKS

SOURCE
UPDATE LIBRARY ON MSS: NSRDCPL,UN=CSYS
OBJECT
EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC,,GETCHR,OUTPUT,MSACCES=<PASSWORD>.

FUNCTION 'GETDABA'

PURPOSE

GET DYNAMIC AREA BASE ADDRESS AND DETERMINE IF CMM IS ACTIVE

FUNCTIONAL CATEGORIES: K2 Q0

LANGUAGE: FORTRAN 77 EXTENDED

COMPUTERS (OPERATING SYSTEMS)

CDC 6000/CYBER 170 (NOS/BE)

REMARKS

NONE

USAGE

INTEGER GETDABA

LOGICAL CMM

...

GETDABA (CMM)

DESCRIPTION OF PARAMETERS

CMM - LOG - WILL CONTAIN EITHER

TRUE - CMM IS ACTIVE

FALSE - CMM IS NOT ACTIVE

GETDABA - INT - WILL RETURN THE ADDRESS OF THE FIRST WORD OF
THE DYNAMIC AREA

CM REQUIRED: 25B

METHOD

THE DABA IS IN WORD RA+65B, BITS 17-0. IF POSITIVE, CMM IS
INACTIVE; IF NEGATIVE, CMM IS ACTIVE.

SUBPROGRAMS REQUIRED

PART OF LANGUAGE

AND SHIFT

OTHERS

MFETCH - READ WORD OF MEMORY

AUTHOR

DAVID V SOMMER - DTNSRDC CODE 1892.2

DATE WRITTEN: 02/16/83

DATE(S) REVISED

LOCATION OF DECKS

SOURCE DECK

UPDATE LIBRARY ON MSS: NSRDCPL,UN=CSYS

OBJECT DECK

EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC,,GETDABA,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'GETFIT'

PURPOSE

GET SPECIFIED FIT ADDRESS

FUNCTIONAL CATEGORIES: Q0

LANGUAGE: CDC 6000 CP COMPASS

COMPUTERS (OPERATING SYSTEMS)

CDC 6000/CYBER 170 (NOS/BE)

REMARKS

NONE

USAGE

CALL GETFIT (LFN, ADDR)

DESCRIPTION OF PARAMETERS

LFN - LOCAL FILE NAME

(LEFT-JUSTIFIED, ZERO-FILLED)

(E.G., 5LTAPE1)

ADDR - WILL CONTAIN THE FIT ADDRESS

CM REQUIRED: 25B

SUBPROGRAMS REQUIRED

PART OF LANGUAGE

NONE

OTHERS

NONE

AUTHOR

ANTHONY CINCOTTA - DTNSRDC CODE 1892.3

DATE WRITTEN: 03/20/75

DATE(S) REVISED

03/23/83 - RECOMPILE AT NOS/BE LEVEL 552

LOCATION OF DECKS

SOURCE

UPDATE LIBRARY ON MSS: NSRDCPL,UN=CSYS

OBJECT

EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC,,GETFIT,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'GETHOUR'

PURPOSE

FOR A SPECIFIED PERIOD OF TIME (UP TO 2 HR 59 MIN 59 SEC)
DETERMINE WHICH HOUR IS OCCUPIED THE LONGEST

FUNCTIONAL CATEGORIES: M2

LANGUAGE: FORTRAN IV

COMPUTERS (OPERATING SYSTEMS)
CDC 6000/CYBER 170 (NOS/BE)

REMARKS
NONE

USAGE

CALL GETHOUR (FROM, TO, HOUR)

DESCRIPTION OF PARAMETERS

FROM - STARTING TIME ('HH.MM.SS ', ' HH.MM.SS ' OR
' HH.MM.SS')
TO - STOPPING TIME (SAME FORMAT AS 'FROM')
HOUR - WILL CONTAIN AN INTEGER HOUR
0 - TIME PERIOD TOO LONG TO DETERMINE HOUR
N - MOST/ALL TIME IS IN THE HOUR N-1 TO N
(E.G., HOUR=8 MEANS MOST/ALL TIME IS IN
THE HOUR 7-8)

CM REQUIRED: 121B

SUBPROGRAMS REQUIRED

PART OF LANGUAGE
AND SHIFT
OTHERS
ISEC - CONVERT HH.MM.SS TO SECONDS

ARITHMETIC STATEMENT FUNCTIONS

I21FMT - FAST I-FORMAT DECODE
L11FMT - FAST L-FORMAT DECODE (LEFT-ADJ, ZERO-FILLED)

METHOD

THE HOURS IN FROM (HF) AND TO (HT) ARE COMPARED.
IF EQUAL, HOUR IS SET TO HT+1.
IF THE DIFFERENCE IS 1, THE AMOUNT OF TIME SPENT IN EACH
HOUR IS COMPARED AND THE HOUR IS SET TO THE LARGER+1.
IF AN EQUAL AMOUNT OF TIME IS SPENT IN EACH HOUR, HOUR IS
SET TO HT+1.
IF THE DIFFERENCE IS 2, HOUR IS SET TO THE MIDDLE HOUR+1.

AUTHOR

DAVID V SOMMER - DTNSRDC CODE 1892.2

DATE WRITTEN: 07/23/76

DATE(S) REVISED

11/16/76

03/23/83 - RECOMPILE AT NOS/BE LEVEL 552

LOCATION OF DECKS

SOURCE

UPDATE LIBRARY ON MSS: NSRDCPL,UN=CSYS

OBJECT

EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC,,GETHOUR,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'GETLFNS'

PURPOSE

GET ACTUAL LOCAL FILE NAMES (FOR FTN)

FUNCTIONAL CATEGORIES: Q0

LANGUAGE: FORTRAN IV EXTENDED

COMPUTERS (OPERATING SYSTEMS)

CDC 6000/CYBER 170 (NOS/BE)

REMARKS

USEFUL ONLY IN FTN PROGRAMS (WHICH ALLOW FILE NAME
REPLACEMENT IN THE 'LGO' CARD).

USAGE

CALL GETLFNS (LFNS, NLFN)
CALL GETLFNS (LFNS)

DESCRIPTION OF PARAMETERS

LFNS - ARRAY DIMENSIONED AT LEAST 1 GREATER THAN NUMBER OF
FILES (INCLUDING EQUATED FILES) ON FTN PROGRAM
STATEMENT

(LFNS(NLFN) WILL BE SET TO 0)

NLFN - IF PRESENT, WILL RETURN NUMBER OF FILE NAMES + 1
(SUBSCRIPT OF FINAL ZERO-WORD IN ARRAY LFNS)

CM REQUIRED: 41B

EXAMPLES

PROGRAM SAMPLE (INPUT,OUTPUT,TAPE1,TAPE5=INPUT,TAPE6=OUTPUT)
DIMENSION LFN(6)
CALL GETLFNS (LFN, NLFN)

...

EXECUTE CARD:	LGO.	LGO,,OUT,TAPE2.
AFTER CALL:	LFN(1) = 5LINPUT	LFN(1) = 5LINPUT
	LFN(2) = 6LOUTPUT	LFN(2) = 3LOUT
	LFN(3) = 5LTAPE1	LFN(3) = 5LTAPE2
	LFN(4) = 5LINPUT	LFN(4) = 5LINPUT
	LFN(5) = 6LOUTPUT	LFN(5) = 3LOUT
	LFN(6) = 0	LFN(6) = 0
	NLFN = 6	NLFN = 6

METHOD

FILE NAMES FROM PROGRAM CARD ARE IN RA+2 ON. EACH HAS A
POINTER TO ITS FIT. THE FIRST WORD OF EACH FIT IS THE
ACTUAL FILE NAME. THE LIST, STARTING IN RA+2, ENDS IN A
WORD OF ZEROS.

SUBPROGRAMS REQUIRED
PART OF LANGUAGE
AND LOCF
OTHERS
NONE

ARITHMETIC STATEMENT FUNCTIONS
L71FMT - FAST L-FORMAT DECODE (LEFT-ADJ, ZERO-FILLED)
R38FMT - FAST R-FORMAT DECODE (RIGHT-ADJ, ZERO-FILLED)

AUTHOR
DAVID V SOMMER - DTNSRDC CODE 1892.2

DATE WRITTEN: 12/30/74

DATE(S) REVISED
12/29/75
10/20/77 - REWRITE TO REDUCE CM REQUIREMENT AND ELIMINATE
SPECIAL FUNCTION CALL
02/15/83 - RECOMPILE AT NOS/BE LEVEL 552

LOCATION OF DECKS
SOURCE
UPDATE LIBRARY ON MSS: NSRDCPL,UN=CSYS
OBJECT
EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT
BEGIN,DOCGET,,NSRDC,,GETLFNS,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'GETLGO'

PURPOSE

EXTRACT FIRST 10 CHARACTERS OF ALL EXECUTE CARD
PARAMETERS

FUNCTIONAL CATEGORIES: Q0

LANGUAGE: FORTRAN IV EXTENDED

COMPUTERS -(OPERATING SYSTEMS)
CDC 6000/CYBER 170 (NOS/BE)

REMARKS
NONE

USAGE
CALL GETLGO (LGO, NLGO)

DESCRIPTION OF PARAMETERS

LGO - ARRAY TO CONTAIN EXECUTE CARD PARAMETERS
LGO(1) CONTAINS EXECUTE NAME
LGO(2)-LGN(NLGO) CONTAIN FIRST 10 CHARACTERS
OF EACH PARAMETER (0 MEANS PARAMETER OMITTED)
NLGO - NUMBER OF WORDS OF LGO FILLED

CM REQUIRED: 26B

METHOD
PARAMETERS ARE EXTRACTED FROM RA+70B THRU RA+77B.

SUBPROGRAMS REQUIRED
PART OF LANGUAGE
NONE
OTHERS
EXPRM - GET NEXT PARAMETER FROM EXECUTE CARD

AUTHOR
DAVID V SOMMER - DTNSRDC CODE 1892.2

DATE WRITTEN: 09/01/77

DATE(S) REVISED
03/23/83 - RECOMPILE AT NOS/BE LEVEL 552

LOCATION OF DECKS
SOURCE
UPDATE LIBRARY ON MSS: NSRDCPL,UN=CSYS
OBJECT
EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT
BEGIN,DOCGET,,NSRDC,,GETLGO,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'GETRA'

PURPOSE

GET FIRST 100B WORDS OF USER'S FL

FUNCTIONAL CATEGORIES: K2

LANGUAGE: CDC 6000 CP COMPASS

COMPUTERS (OPERATING SYSTEMS)

CDC 6000/CYBER 170 (NOS/BE)

REMARKS

NONE

USAGE

CALL GETRA (RA)

DESCRIPTION OF PARAMETER

RA - 64-WORD ARRAY TO HOLD FIRST 100B WORDS OF FL

CM REQUIRED: 7B

SUBPROGRAMS REQUIRED

PART OF LANGUAGE

NONE

OTHERS

NONE

AUTHOR

DAVID V SOMMER - DTNSRDC CODE 1892.2

DATE WRITTEN: 10/03/73

DATE(S) REVISED

03/23/83 - RECOMPILE AT NOS/BE LEVEL 552

LOCATION OF DECKS

SOURCE

UPDATE LIBRARY ON MSS: NSRDCPL,UN=CSYS

OBJECT

EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC,,GETRA,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'GODROP'

PURPOSE

CREATE GO/DROP MESSAGE AND PROCESS RESPONSE

FUNCTIONAL CATEGORIES: Q0

LANGUAGE: FORTRAN IV EXTENDED

COMPUTERS (OPERATING SYSTEMS)

CDC 6000/CYBER 170 (NOS/BE)

REMARKS

NOT DESIGNED FOR BATCH JOBS.

IN INTERCOM, WILL GENERATE MESSAGE AT THE TERMINAL, NOT AT
THE CENTRAL SITE CONSOLE.

WHEN USED WITH NO ARGUMENT LIST, THE 'Z' PARAMETER MUST BE
USED ON THE FTN CARD.

USAGE

CALL GODROP (MESSAGE)

CALL GODROP

DESCRIPTION OF PARAMETER

MESSAGE - IF USED, CONTENTS WILL BE DISPLAYED (SHOULD BE A
ZERO-BYTE TERMINATED FIELD)

IF OMITTED, THE MESSAGE IS TAKEN FROM RA+70B THRU
RA+77B AND PREFIXED WITH 'GO/DROP- '. THE
MESSAGE MAY BE INSERTED BY

'CALL PUTRA (MESSAGE, 70B, 76B)'

CM REQUIRED: 142B

SUBPROGRAMS REQUIRED

PART OF LANGUAGE

LOCF REMARK

OTHERS

MFETCH - READ A WORD IN USER'S FL

MSET - SET WORD IN USER'S FL

AUTHOR

C FLICK - KPS NWL

DATE WRITTEN: 06/73

DATE(S) REVISED

03/23/83 - RECOMPILE AT NOS/BE LEVEL 552

LOCATION OF DECKS

SOURCE

UPDATE LIBRARY ON MSS: NSRDCPL,UN=CSYS

OBJECT

EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC,,GODROP,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'HELP'

PURPOSE

COMPLEX ROOTS OF A REAL OR COMPLEX POLYNOMIAL

FUNCTIONAL CATEGORIES: C2 A2

LANGUAGE: FORTRAN IV EXTENDED

COMPUTERS (OPERATING SYSTEMS)
CDC 6000/CYBER 170 (NOS/BE)

REMARKS

CALCULATES THE ROOTS OF THE COMPLEX POLYNOMIAL

$FN(Z) = A(N)*Z^{**}(N) + A(N-1)*Z^{**}(N-1) + \dots + A(1)*Z + A(0)$
WHERE $A(I)$ ($I=0,1,\dots,N$) ARE PSEUDO-COMPLEX COEFFICIENTS.

SEE ALSO MSL SUBROUTINE 'HELP'.

USAGE

CALL HELP (N, A, ROOT, TAU, ETAI, MI)

DESCRIPTION OF PARAMETERS

N - DEGREE OF POLYNOMIAL
(DESTROYED BY HELP)
A - ARRAY OF N+1 COEFFICIENTS (SEE NOTE)
(DESTROYED BY HELP)
ROOT - ARRAY TO CONTAIN THE N ROOTS (SEE NOTE)
TAU - THE TOLERANCE TO BE PRESCRIBED FOR $FN(ROOT(I))$
($ROOT(I)$ WOULD BE CONSIDERED AS A ROOT WHEN
 $ABS(FN(ROOT(I))) \leq TAU$
IN THE SCALE OF THE SYSTEM OF COORDINATES
CONSIDERED AT THE MOMENT)
ETAI - INDICATOR ARRAY
ETAI(I)=+1 -- $ABS(FN(ZI)) \leq TAU$
= 0 -- DID NOT FIND A NEW CIRCLE
=-1 -- INCREMENTING THE ROOT BY NU DID NOT
CHANGE THE ROOT (BECAUSE OF MACHINE
LIMITS)
MI - INDICATOR VECTOR

NOTE: ARRAYS 'A' AND 'ROOT' ARE 2-DIMENSIONAL REAL ARRAYS
 $A(N+1,2)$, $ROOT(N,2)$, WHERE $A(I,1)$, $ROOT(I,1)$ ARE THE
REAL PARTS AND $A(I,2)$, $ROOT(I,2)$ ARE THE IMAGINARY
PARTS.

CM REQUIRED: 1400B

METHOD

THE METHOD OF D. H. LEHMER (JOURNAL ACM, 1961, VOL 8,
P. 151) IS USED.

SUBPROGRAMS REQUIRED

PART OF LANGUAGE

ABS SIN SQR T

PART OF PROGRAM

ANSWER

ANULUS

COMADD

COMMUL

DIVIDE

FUNC

OVLAP

OTHERS

NONE

AUTHORS

ADEL S. ABDELGAWAD

G. MIEDEL

DEUTSCHES RECHENZENTRUM

SHARE PROGRAM NUMBER 3400

DATE WRITTEN: 11/64

DATE(S) REVISED

11/18/65 - GM

09/20/83 - RECOMPILE AT NOS/BE LEVEL 552

LOCATION OF DECKS

SOURCE

UPDATE LIBRARY ON MSS: NSRDCPM,UN=CSYS (*DECK ZFHELP)

OBJECT

EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC,,HELP,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'HERE'
FUNCTION 'HERE'

PURPOSE

GET TERMINAL ID FOR THIS JOB

FUNCTIONAL CATEGORIES: Q0

LANGUAGE: FORTRAN IV EXTENDED

COMPUTERS (OPERATING SYSTEMS)
CDC 6000/CYBER 170 (NOS/BE)

REMARKS

NONE

USAGE

CALL HERE (I)
VARIABLE = HERE (I)

DESCRIPTION OF PARAMETERS

I - WILL CONTAIN THE TERMINAL ID, LEFT-JUSTIFIED,
ZERO-FILLED (1LC = CENTRAL SITE)
(WHEN USED AS A FUNCTION, 'HERE' WILL CONTAIN THE SAME AS
'I'. 'VARIABLE' AND 'HERE' MUST BE OF THE SAME TYPE.)

CM REQUIRED: 40B

METHOD

THE TERMINAL ID IS TAKEN FROM CONTROL POINT AREA.
IF THIS FIELD IS ZERO, IT IS A CENTRAL SITE JOB. IN THIS
CASE, 1LC IS RETURNED.

SUBPROGRAMS REQUIRED

PART OF LANGUAGE

AND SHIFT

OTHERS

RCPA - READ CONTROL POINT AREA

UNHEX3 - CONVERT 2-CHAR DISPLAY CODE TO 3-CHAR HEX

ARITHMETIC STATEMENT FUNCTIONS

L25FMT - FAST L-FORMAT DECODE (LEFT-ADJ, ZERO-FILLED)

AUTHOR

DAVID V SOMMER - DTNSRDC CODE 1892.2

DATE WRITTEN: 12/05/75

DATE(S) REVISED

10/01/78 - CHANGE FOR 3-CHARACTER TERMINAL ID

03/21/83 - RECOMPILE AT NOS/BE LEVEL 552

LOCATION OF DECKS

SOURCE

UPDATE LIBRARY ON MSS: NSRDCPL,UN=CSYS

OBJECT

EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC,,HERE,OUTPUT,MSACCES=<PASSWORD>.

FUNCTION 'HEX3'

PURPOSE

SQUEEZE 3-CHARACTER HEX INTO 12 BITS

FUNCTIONAL CATEGORIES: M2

LANGUAGE: FORTRAN IV EXTENDED

COMPUTERS (OPERATING SYSTEMS)

CDC 6000/CYBER 170 (NOS/BE)

REMARKS

'HEX3' IS AN INTEGER FUNCTION.

WRITTEN TO CHANGE USER-SUPPLIED 3-CHARACTER HEX TERMINAL ID
TO THE FORM NEEDED BY THE CALLABLE ROUTE.

USAGE

INTEGER HEX

I = HEX3 (HEXVAL)

DESCRIPTION OF PARAMETERS

HEXVAL - INPUT HEX VALUE (E.G., 3LF04)

HEX3 - OUTPUT IN FIRST 2 CHARACTERS (E.G., 2L@D)

CM REQUIRED: 61B

SUBPROGRAMS REQUIRED

PART OF LANGUAGE

AND

OR

SHIFT

OTHERS

NONE

ARITHMETIC STATEMENT FUNCTIONS

FAST R-FORMAT DECODE (RIGHT-ADJ, ZERO-FILLED)

R11FMT R12FMT R13FMT

AUTHOR

DAVID V SOMMER - DTNSRDC CODE 1892.2

DATE WRITTEN: 09/19/78

DATE(S) REVISED

03/21/83 - RECOMPILE AT NOS/BE LEVEL 552

LOCATION OF DECKS

SOURCE

UPDATE LIBRARY ON MSS: NSRDCPL,UN=CSYS

OBJECT

EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC,,HEX3,OUTPUT,MSACCES=<PASSWORD>.

FUNCTION 'IAOC'

PURPOSE

COUNT ONE-BITS IN SPECIFIED WORD

FUNCTIONAL CATEGORIES: G6

LANGUAGE: CDC 6000 CP COMPASS

COMPUTERS (OPERATING SYSTEMS)

CDC 6000/CYBER 170 (NOS/BE)

REMARKS

NONE

USAGE

N = IAOC (I)

DESCRIPTION OF PARAMETERS

I - WORD TO BE PROCESSED

IAOC - NUMBER OF ONE-BITS

CM REQUIRED: 2B

SUBPROGRAMS REQUIRED

PART OF LANGUAGE

NONE

OTHERS

NONE

AUTHOR

FROM NWL

DATE WRITTEN:

DATE(S) REVISED

LOCATION OF DECKS

SOURCE

UPDATE LIBRARY ON MSS: NSRDCPL,UN=CSYS

OBJECT

EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC,,IAOC,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'IBL'
FUNCTION 'IBL'

PURPOSE

CALCULATE BEST BLOCK LENGTH (MIN TIME REQ'D FOR RANDOM
ACCESS AND MINIMUM BUFFER SIZE) FOR INDEX SEQUENTIAL FILES

FUNCTIONAL CATEGORIES: Q0

LANGUAGE: FORTRAN IV

COMPUTERS (OPERATING SYSTEMS)
CDC 6000/CYBER 170 (NOS/BE)

REMARKS

THIS SUBROUTINE CALCULATES BEST BLOCK LENGTHS FOR INDEX SEQ
FILES BASED ON EITHER VALUES ESTABLISHED IN A FIT OR A SIX
WORD TABLE. IF INPUT IS A FIT, THIS ROUTINE WILL SET FIT
FIELDS MBL AND IBL TO THE VALUE IT CALCULATES. A SHORT
(5 LINE) REPORT CAN BE PRINTED DEPENDING ON THE VALUE OF
THE SECOND PARAMETER PASSED TO IBL.

SEVERAL ASSUMPTIONS ARE MADE IN DERIVING THE FORMULA THIS
SUBROUTINE USES. AMONG THESE ARE:

1. GENERAL INDEX-SEQ PROCESSING IS ASSUMED.
IF THE FILE IS PROCESSED RANDOMLY ONLY, FILE ORGANIZA-
TIONS OTHER THAN INDEX-SEQ PROVIDE BETTER PERFORMANCE.
IF THE FILE IS ACCESSED HEAVILY SEQUENTIALLY THIS
CALCULATION MAY NOT PROVIDE THE OPTIMUM SIZE.
2. EQUAL LENGTH DATA AND INDEX BLOCKS ARE ASSUMED TO ALLOW
SHARING OF BUFFER AREAS.
3. BLOCK SIZE SHOULD BE OF MINIMAL LENGTH WHICH ALLOWS THE
FILE TO BE FILLED TO CAPACITY INCLUDING PADDING.
4. BUFFER SPACE IS KEPT NEARLY MINIMAL AND RANDOM ACCESS
TIME IS KEPT NEARLY MINIMAL.

THE ROUTINE IS BASED ON AN ARTICLE PUBLISHED IN CONTROL
DATA PSI EXCERPTS (NO. 109 - OCTOBER 1977).

THE ROUTINE CANNOT BE USED IF RESULTING BLOCK LENGTH IS
SMALLER THAN MAX REC LENGTH. IT SHOULD NOT BE USED IF
RECORD TRUNCATION RESULTS IN EXCESSIVE PADDING IN THE
DATA BLOCKS.

THE TIMINGS IN THE OUTPUT REPORT ARE BASED ON THE
ASSUMPTION OF:

ACCESS TIME (POSITION + LATENCY) = 30 MS
TRANSFER TIME = 1 MS/PRU
CP TIME TO PROCESS THE REQUEST = 1 MS

SO TOTAL TIME = 1 + (NO. INDEX LEVELS)*(30+NPRUS)

USAGE

FORTRAN CALLING SEQUENCES

CALL IBL (FIT, IFLAG)

IBLKSZ = IBL (FIT, IFLAG)

COBOL CALLING SEQUENCE

ENTER IBL USING FIT, IFLAG.

FIT - FILE INFORMATION TABLE -OR-

A SIX-WORD INTEGER ARRAY CONTAINING:

FLM MAX RECORDS IN THE FILE

RL AVERAGE RECORD LENGTH

KL KEY LENGTH

IP INDEX PADDING PERCENT

DP DATA PADDING PERCENT

MRL MAX RECORD LENGTH

IFLAG - PRINTOUT FLAG

"Y" - PRINT 5-LINE REPORT

OTHER - DO NOT PRINT

IBL - WHEN USED AS A FORTRAN FUNCTION, IBL RETURNS
THE COMPUTED BLOCK SIZE

CM REQUIRED: 342B

SUBPROGRAMS REQUIRED

PART OF LANGUAGE

IFETCH STOREF

OTHERS

NONE

AUTHOR

ACQUIRED FROM AUTHOR OF CDC PSI ARTICLE

MODIFIED BY BRUCE D. BLACK - DTNSRDC CODE 1892.1 (CDC)

DATE WRITTEN: 04/03/78

DATE(S) REVISED

04/03/78 - ADD OPTION TO TURN OFF PRINT OF REPORT

03/23/83 - RECOMPILE AT NOS/BE LEVEL 552

LOCATION OF DECKS

SOURCE

UPDATE LIBRARY ON MSS: NSRDCPL,UN=CSYS

OBJECT

EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC,,IBL,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'IBUNP'

PURPOSE

UNPACK 12-BIT BYTES FROM ARRAY

FUNCTIONAL CATEGORIES: M4

LANGUAGE: CDC 6000 CP COMPASS

COMPUTERS (OPERATING SYSTEMS)

CDC 6000/CYBER 170 (NOS/BE)

REMARKS

NONE

USAGE

CALL IBUNP (A1, A2, N)

DESCRIPTION OF PARAMETERS

A1 - INPUT ARRAY FROM WHICH BYTES ARE UNPACKED

A2 - OUTPUT ARRAY INTO WHICH BYTES ARE PLACED,
1 BYTE PER WORD, RIGHT JUSTIFIED, WITH LEADING ZEROS

N - NUMBER OF CDC WORDS TO UNPACK

DIMENSION OF A1 IS N

DIMENSION OF A2 IS 5*N

CM REQUIRED: 12B

SUBPROGRAMS REQUIRED

PART OF LANGUAGE

NONE

OTHERS

NONE

AUTHOR

FROM NWL

DATE WRITTEN:

DATE(S) REVISED

LOCATION OF DECKS

SOURCE

UPDATE LIBRARY ON MSS: NSRDCPL,UN=CSYS

OBJECT

EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC,,IBUNP,OUTPUT,MSACCES=<PASSWORD>.

FUNCTION 'IDAYWEK'

PURPOSE

DETERMINE THE DAY OF THE WEEK FOR ANY DATE FROM 10/15/1582
THRU 02/28/4000

FUNCTIONAL CATEGORIES: G6

LANGUAGE: FORTRAN IV EXTENDED

COMPUTERS (OPERATING SYSTEMS)
CDC 6000/CYBER 170 (NOS/BE)

REMARKS
NONE

USAGE

IDAY = IDAYWEK (IDATE, ICENT)
IDAY = IDAYWEK (IDATE)

DESCRIPTION OF PARAMETERS

IDATE - DATE TO BE PROCESSED ('MM/DD/YY ' OR ' MM/DD/YY '
OR ' MM/DD/YY')
(IF IDATE = 0, TODAY'S DATE WILL BE USED; IDATE
WILL BE SET TO TODAY'S DATE ' MM/DD/YY ')

ICENT - CENTURY (E.G., 1900)
IF OMITTED, 1900 IS ASSUMED.

IDAYWEK - WILL CONTAIN THE DAY OF THE WEEK IN A-FORMAT
(E.G., 'SUNDAY ')

CM REQUIRED: 104B

SUBPROGRAMS REQUIRED

PART OF LANGUAGE
AND DATE LOCF SHIFT
OTHERS
WEKDAY - DETERMINE DAY OF WEEK

ARITHMETIC STATEMENT FUNCTIONS

FAST I-FORMAT DECODE
I21FMT I24FMT I27FMT
FAST L-FORMAT DECODE (LEFT-ADJ, ZERO-FILLED)
L11FMT

AUTHOR

DAVID V SOMMER - DTNSRDC CODE 1892.2

DATE WRITTEN: 04/06/77

DATE(S) REVISED

03/23/83 - RECOMPILE AT NOS/BE LEVEL 552

LOCATION OF DECKS

SOURCE

UPDATE LIBRARY ON MSS: NSRDCPL,UN=CSYS

OBJECT

EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC,,IDAYWEK,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'IDID'
FUNCTION 'IDID'

PURPOSE

GET USER INITIALS (AND INTERCOM USER ID) FROM CHARGE CARD
OR LOGIN

FUNCTIONAL CATEGORIES: Q0

LANGUAGE: FORTRAN IV EXTENDED

COMPUTERS (OPERATING SYSTEMS)
CDC 6000/CYBER 170 (NOS/BE)

REMARKS

IF USER INITIALS AND USER ID ARE EQUAL, IT IS A BATCH JOB.

USAGE

CALL IDID (ID, IUSERID)
CALL IDID (ID)
IID = IDID (ID, IUSERID)
IID = IDID (ID)

DESCRIPTION OF PARAMETERS

ID - WILL CONTAIN 4-CHARACTER USER INITIALS FROM
CHARGE CARD OR START OF LOGIN

IUSERID - WILL CONTAIN 4-CHARACTER USER INITIALS FROM
CHARGE CARD OR UP TO 10-CHARACTER USER ID
FROM LOGIN

(IF ID = IUSERID, IT IS A BATCH JOB)

WHEN USED AS A FUNCTION, THE CONTENTS OF ID IS ALSO RETURNED
AS THE FUNCTION VALUE.

CM REQUIRED: 27B

METHOD

THE ID IS TAKEN FROM THE CONTROL POINT AREA.

SUBPROGRAMS REQUIRED

PART OF LANGUAGE
LOCF

OTHERS

RCPA - READ CONTROL POINT AREA

ARITHMETIC STATEMENT FUNCTIONS

L41FMT - FAST L-FORMAT DECODE (LEFT-ADJ, ZERO-FILLED)

AUTHOR

DAVID V SOMMER - DTNSRDC CODE 1892.2

DATE WRITTEN: 01/28/77

DATE(S) REVISED

03/23/83 - RECOMPILE AT NOS/BE LEVEL 552

LOCATION OF DECKS

SOURCE

UPDATE LIBRARY ON MSS: NSRDCPL,UN=CSYS

OBJECT

EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC,,IDID,OUTPUT,MSACCES=<PASSWORD>.

FUNCTION 'IDIGIT'

PURPOSE

CHECK FOR DIGITS IN A FIELD WITHIN A WORD

FUNCTIONAL CATEGORIES: M5

COMPUTERS (OPERATING SYSTEMS)

CDC 6000/CYBER 170 (NOS/BE)

REMARKS

THE VALUE RETURNED IS ONE OF THE FOLLOWING:

- 11 - ERROR - ISTOP < ISTART
 - N - ERROR - START NON-DIGIT FOUND IN POSITION N
 - 0 - ALL POSITIONS IN FIELD ARE DIGITS
 - +N - OOB FOUND IN POSITION N
- ALL PRECEDING CHARACTERS ARE DIGITS

USAGE

IDIGIT (I, ISTART, ISTOP)

IDIGIT (I, ISTART)

IDIGIT (I)

DESCRIPTION OF PARAMETERS

- I - WORD TO BE ANALYZED
- ISTART - STARTING POSITION OF FIELD TO BE CHECKED
(1-10, DEFAULT: 1)
- ISTOP - STOP POSITION OF FIELD TO BE CHECKED
(1-10, DEFAULT: 10)
(TESTING WILL STOP IF OOB ENCOUNTERED)

CM REQUIRED: 50B

SUBPROGRAMS REQUIRED

PART OF LANGUAGE

LOCF MAXO MINO SHIFT

OTHERS

NONE

AUTHOR

DAVID V SOMMER - DTNSRDC CODE 1892.2

DATE WRITTEN: 05/13/75

DATE(S) REVISED

03/23/83 - RECOMPILE AT NOS/BE LEVEL 552

LOCATION OF DECKS

SOURCE

UPDATE LIBRARY ON MSS: NSRDCPL,UN=CSYS

OBJECT

EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC,,IDIGIT,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'IFINDCH'
FUNCTION 'IFINDCH'

PURPOSE

FIND FIRST OCCURRENCE OF SPECIFIED CHARACTER IN ARRAY

FUNCTIONAL CATEGORIES: M5

LANGUAGE: FORTRAN IV

COMPUTERS (OPERATING SYSTEMS)
CDC 6000/CYBER 170 (NOS/BE)

REMARKS
NONE

USAGE

CALL IFINDCH (A, NA, CHAR, NC, NW)
NC = IFINDCH (A, NA, CHAR, NC, NW)
NC = IFINDCH (A, NA, CHAR)

DESCRIPTION OF PARAMETERS

A - ARRAY TO BE SEARCHED
NA - NUMBER OF WORDS IN 'A' TO BE SEARCHED
CHAR - CHARACTER TO BE SEARCHED FOR (1R FORMAT)
NC - OUTPUT POSITION OF FIRST OCCURRENCE OF CHAR IN 'A'
(IF NO MATCH, ZERO (0) IS RETURNED)
NW - OUTPUT SUBSCRIPT OF WORD IN 'A' CONTAINING CHAR
(IF NO MATCH, NW IS SET TO NA)

CM REQUIRED: 60B

SUBPROGRAMS REQUIRED

PART OF LANGUAGE
LOCF

OTHERS

GETCHA - GET CHARACTER FROM ARRAY

AUTHOR

DAVID V SOMMER - DTNSRDC CODE 1892.2

DATE WRITTEN: 04/20/76

DATE(S) REVISED

11/02/76 - CHANGE TO FUNCTION AND SUBROUTINE
02/15/83 - RECOMPILE AT NOS/BE LEVEL 552

LOCATION OF DECKS

SOURCE

UPDATE LIBRARY ON MSS: NSRDCPL,UN=CSYS

OBJECT

EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC,,IFINDCH,OUTPUT,MSACCES=<PASSWORD>.

FUNCTION 'IFMTV'

PURPOSE

FAST I-FORMAT DECODE OF VARIABLE LENGTH INPUT (UNSIGNED,
POSITIVE INTEGER)

LANGUAGE: FORTRAN IV EXTENDED

COMPUTERS (OPERATING SYSTEMS)
CDC 6000/CYBER 170 (NOS/BE)

FUNCTIONAL CATEGORIES: 14

REMARKS

USEFUL IN DECODING INTEGERS PASSED AS ARGUMENTS IN THE
EXECUTE STATEMENT FOR A FTN PROGRAM.

USAGE

IFMTV (I)

DESCRIPTION OF PARAMETER

I - SINGLE WORD CONTAINING NUMBER TO BE DECODED;
1-10 DIGITS, LEFT-JUSTIFIED, ZERO-PADDED;
A NON-DIGIT EMBEDDED IN THE FIELD WILL RETURN -1
(EG, 3L987 WILL RETURN THE INTEGER 987;
6L9 7654 WILL RETURN -1 (EMBEDDED BLANK))

CM REQUIRED: 30B

SUBPROGRAMS REQUIRED

PART OF LANGUAGE
SHIFT
OTHERS
NONE

AUTHOR

DAVID V SOMMER - DTNSRDC CODE 1892.2

DATE WRITTEN: 10/74

DATE(S) REVISED

02/15/83 - RECOMPILE AT NOS/BE LEVEL 552

LOCATION OF DECKS

SOURCE
UPDATE LIBRARY ON MSS: NSRDCPL,UN=CSYS
OBJECT
EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC,,IFMTV,OUTPUT,MSACCES=<PASSWORD>.

FUNCTION 'IHMS'

PURPOSE

CONVERT SECONDS TO ' HH.MM.SS.'

FUNCTIONAL CATEGORIES: M2

LANGUAGE: FORTRAN IV

COMPUTERS (OPERATING SYSTEMS)

CDC 6000/CYBER 170 (NOS/BE)

REMARKS

NONE

USAGE

IHMS (ISEC)

DESCRIPTION OF PARAMETER

ISEC - TIME (IN SECONDS) TO BE CONVERTED

CM REQUIRED: 44B

SUBPROGRAMS REQUIRED

PART OF LANGUAGE

OR SHIFT

OTHERS

NONE

AUTHOR

DAVID V SOMMER - DTNSRDC CODE 1892.2

DATE WRITTEN: 05/08/74

DATE(S) REVISED

03/23/83 - RECOMPILE AT NOS/BE LEVEL 552

LOCATION OF DECKS

SOURCE

UPDATE LIBRARY ON MSS: NSRDCPL,UN=CSYS

OBJECT

EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC,,IHMS,OUTPUT,MSACCES=<PASSWORD>.

FUNCTION 'IPAKLFT'

PURPOSE

SQUEEZE LEFT AND REMOVE ZEROS (00B) AND BLANKS (55B), RETURN
NUMBER OF CHARACTERS

FUNCTIONAL CATEGORIES: M4

LANGUAGE: FORTRAN IV EXTENDED

COMPUTERS (OPERATING SYSTEMS)
CDC 6000/CYBER 170 (NOS/BE)

REMARKS

IF ANY BLANKS OR ZEROS WERE REMOVED, THE ARRAY IS PADDED
WITH TRAILING ZEROS

USAGE

NCHAR = IPAKLFT (A)
NCHAR = IPAKLFT (A, NA)

DESCRIPTION OF PARAMETERS

A - ARRAY TO BE PROCESSED
NA - NUMBER OF WORDS TO BE PROCESSED
(OMITTED = 1)
IPAKLFT - NUMBER OF NON-BLANK (NON-ZERO) CHARACTERS AFTER
PROCESSING

CM REQUIRED: 72B

EXAMPLE

DIMENSION A(3)
DATA A/ "THIS IS A SAMPLE FIELD"/
NCHAR = IPAKLFT (A, 3)
...
AFTER EXECUTION: 'A' = 18LTHISISASAMPLEFIELD, IPAKLFT = 18

SUBPROGRAMS REQUIRED

PART OF LANGUAGE
LOCF

OTHERS

GETCHA - GET A CHARACTER
PUTCHA - PUT A CHARACTER

AUTHOR

DAVID V SOMMER - DTNSRDC CODE 1892.2

DATE WRITTEN: 07/25/77

DATE(S) REVISED

03/21/83 - RECOMPILE AT NOS/BE LEVEL 552

LOCATION OF DECKS

SOURCE

UPDATE LIBRARY ON MSS: NSRDCPL,UN=CSYS

OBJECT

EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC,,IPAKLFT,OUTPUT,MSACCES=<PASSWORD>.

FUNCTION 'IROMAN'

PURPOSE

CONVERT ROMAN NUMBERS TO INTEGER

FUNCTIONAL CATEGORIES: M2

LANGUAGE: FORTRAN IV

COMPUTERS (OPERATING SYSTEMS)

CDC 6000/CYBER 170 (NOS/BE)

REMARKS

VALIDITY OF THE ROMAN NUMBER IS NOT CHECKED. INVALID ROMAN
NUMERALS ARE IGNORED. ROMAN NUMBER ENDS WHEN FIRST
BLANK OR OOB IS ENCOUNTERED.

USAGE

IVAR = IROMAN (NUMBER)

DESCRIPTION OF PARAMETERS

IROMAN - WILL CONTAIN INTEGER EQUIVALENT OF SUPPLIED
ROMAN NUMBER

NUMBER - ROMAN NUMBER TO BE CONVERTED

CM REQUIRED: 130B

EXAMPLES

MCMLXXVI WILL RETURN THE INTEGER 1976

I " " " " 1

ETC.

SUBPROGRAMS REQUIRED

PART OF LANGUAGE

NONE

OTHERS

GETCHA - EXTRACT CHARACTER FROM AN ARRAY

AUTHOR

DAVID V SOMMER - DTNSRDC CODE 1892.2

DATE WRITTEN: 12/02/76

DATE(S) REVISED

03/23/83 - RECOMPILE AT NOS/BE LEVEL 552

LOCATION OF DECKS

SOURCE

UPDATE LIBRARY ON MSS: NSRDCPL,UN=CSYS

OBJECT

EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC,,IROMAN,OUTPUT,MSACCES=<PASSWORD>.

FUNCTION 'ISEC'

PURPOSE

CONVERT HH.MM.SS TO SECONDS

FUNCTIONAL CATEGORIES: Q0

LANGUAGE: FORTRAN IV

COMPUTERS (OPERATING SYSTEMS)

CDC 6000/CYBER 170 (NOS/BE)

REMARKS

NONE

USAGE

ISEC (ITIME)

DESCRIPTION OF PARAMETER

ITIME - TIME TO BE CONVERTED

(MAY BE 'HH.MM.SS. ', ' HH.MM.SS.', OR ' HH.MM.SS')

CM REQUIRED: 40B

SUBPROGRAMS REQUIRED

PART OF LANGUAGE

AND SHIFT

OTHERS

NONE

ARITHMETIC STATEMENT FUNCTIONS

FAST I-FORMAT DECODE

I21FMT I24FMT I27FMT

FAST L-FORMAT DECODE (LEFT-ADJ, ZERO-FILLED)

L11FMT

AUTHOR

DAVID V SOMMER - DTNSRDC CODE 1892.2

DATE WRITTEN: 05/01/74

DATE(S) REVISED

03/02/83 - RECOMPILE AT NOS/BE LEVEL 552

LOCATION OF DECKS

SOURCE

UPDATE LIBRARY ON MSS: NSRDCPL,UN=CSYS

OBJECT

EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC,,ISEC,OUTPUT,MSACCES=<PASSWORD>.

FUNCTION 'ISITCNF'

PURPOSE

SEE IF SPECIFIED FILE IS CONNECTED

FUNCTIONAL CATEGORIES: Q0

LANGUAGE: FORTRAN IV EXTENDED

COMPUTERS (OPERATING SYSTEMS)

CDC 6000/CYBER 170 (NOS/BE)

REMARKS

THE FILE BEING TESTED MUST BE OPENED BEFORE USING
THIS FUNCTION. FOR FORTRAN LFN'S, THIS IS ACCOMPLISHED
BY ANY I/O OPERATION OR CALL CONNEC OR CALL DISCONT.

USAGE

ISITCNF (I)

DESCRIPTION OF PARAMETERS

LFN - FILE TO BE CHECKED (E.G., 5LTAPE1)

ISITCNF - WILL RETURN ONE OF:

+1 - FILE IS CONNECTED

0 - FILE IS NOT CONNECTED

-1 - ERROR - FILE NOT FOUND

-2 - ERROR - LFN = 0

CM REQUIRED: 66B

METHOD

BIT 19 OF WORD 17 (18TH WORD) OF FIT IS EXTRACTED.

SUBPROGRAMS REQUIRED

PART OF LANGUAGE

AND SHIFT

OTHERS

LOFFTN - GET FORTRAN LIST-OF-FILES

MFETCH - GET SPECIFIED WORD IN USER'S FIELD LENGTH

AUTHOR

DAVID V SOMMER - DTNSRDC CODE 1892.2

DATE WRITTEN: 05/02/75

DATE(S) REVISED

12/11/81 - UPGRADE TO LEVEL 508

03/23/83 - RECOMPILE AT NOS/BE LEVEL 552

LOCATION OF DECKS

SOURCE

UPDATE LIBRARY ON MSS: NSRDCPL,UN=CSYS

OBJECT

EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC,,ISITCNF,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'ISSORT'

PURPOSE

FTN-CALLABLE SHELL SORT FOR INTEGER ARRAYS

FUNCTIONAL CATEGORIES: M1

LANGUAGE: FORTRAN IV EXTENDED

COMPUTERS (OPERATING SYSTEMS)

CDC 6000/CYBER 170 (NOS/BE)

REMARKS

THIS MAY BE USED WHEN YOU HAVE DEFINED TWO ARRAYS WHICH
"GO TOGETHER" AND YOU ARE SORTING ONE ARRAY AND WISH TO KEEP
CORRESPONDING ELEMENTS OF THE OTHER ARRAY WITH IT. THAT IS,
WHEREVER A(1) ENDS UP AFTER THE SORT, T(1) WILL BE IN THE
SAME RELATIVE POSITION.

USE SSORT FOR REALS.

USAGE

CALL ISSORT (A, I, T)
CALL ISSORT (A, I)

DESCRIPTION OF PARAMETERS

A - INTEGER ARRAY TO BE SORTED
I - NUMBER OF ELEMENTS TO BE SORTED
T - IF PRESENT, AN ASSOCIATED ARRAY RE-ORDERED TO MAINTAIN
1 TO 1 CORRESPONDENCE WITH THE ELEMENTS OF ARRAY 'A'

CM REQUIRED: 71B

SUBPROGRAMS REQUIRED

PART OF LANGUAGE
LOCF SHIFT
OTHERS
NONE

AUTHORS

C FLINK - KPS NWL
DAVID V SOMMER - DTNSRDC CODE 1892.2

DATE WRITTEN: 12/07/70 (CF - SSORT)

DATE(S) REVISED

10/06/81 - DVS - MAKE INTEGER VERSION OF SSORT
03/23/83 - RECOMPILE AT NOS/BE LEVEL 552

LOCATION OF DECKS

SOURCE
UPDATE LIBRARY ON MSS: NSRDCPL,UN=CSYS
OBJECT
EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC,,ISSORT,OUTPUT,MSACCES=<PASSWORD>.

FUNCTION 'ISTAPE'

PURPOSE

GENERATE TAPE NAME 'TAPENN'

FUNCTIONAL CATEGORIES: M4

LANGUAGE: CDC 6000 CP COMPASS

COMPUTERS (OPERATING SYSTEMS)

CDC 6000/CYBER 170 (NOS/BE)

REMARKS

NONE

USAGE

NAME = ISTAPE (NN)

DESCRIPTION OF PARAMETERS

NAME - RESULTANT DISPLAY CODE NAME 'TAPENN'

(LEFT-JUSTIFIED, ZERO-FILLED)

(5LTAPEN OR 6LTAPENN)

NN - FORTRAN LOGICAL UNIT NUMBER

CM REQUIRED: 23B

SUBPROGRAMS REQUIRED

PART OF LANGUAGE

NONE

OTHERS

NONE

AUTHOR

NWL

DATE WRITTEN: ?

DATE(S) REVISED

03/22/83 - RECOMPILE AT NOS/BE LEVEL 552

LOCATION OF DECKS

SOURCE

UPDATE LIBRARY ON MSS: NSRDCPL,UN=CSYS

OBJECT

EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC,,ISTAPE,OUTPUT,MSACCES=<PASSWORD>.

FUNCTION 'ISUMIT'

PURPOSE

SUM ELEMENTS OF INTEGER ARRAY

FUNCTIONAL CATEGORIES: A1

LANGUAGE: FORTRAN IV

COMPUTERS (OPERATING SYSTEMS)

CDC 6000/CYBER 170 (NOS/BE)

REMARKS

NONE

USAGE

ITOTAL = ISUMIT (IARRAY, N)

DESCRIPTION OF PARAMETERS

ISUMIT - WILL CONTAIN IARRAY(1)+IARRAY(2)+...+IARRAY(N)

IARRAY - ARRAY TO BE SUMMED

N - NUMBER OF ELEMENTS OF IARRAY TO BE SUMMED

CM REQUIRED: 16B

SUBPROGRAMS REQUIRED

PART OF LANGUAGE

NONE

OTHERS

NONE

AUTHOR

DAVID V SOMMER - DTNSRDC CODE 1892.2

DATE WRITTEN: 11/23/76

DATE(S) REVISED

03/23/83 - RECOMPILE AT NOS/BE LEVEL 552

LOCATION OF DECKS

SOURCE

UPDATE LIBRARY ON MSS: NSRDCPL,UN=CSYS

OBJECT

EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC,,ISUMIT,OUTPUT,MSACCES=<PASSWORD>.

FUNCTION 'ITL'

PURPOSE

GET CURRENT INTERCOM TIME LIMIT (ETL)

FUNCTIONAL CATEGORIES: Q0

LANGUAGE: FORTRAN 77

COMPUTERS (OPERATING SYSTEMS)

CDC 6000/CYBER 170 (NOS/BE)

REMARKS

IF THE VALUE RETURNED IS ZERO, THE DEFAULT SETTING IS
IMPLIED (MFE=60 SECONDS; MFF=20 SECONDS).

USAGE

ITL ()

DESCRIPTION OF PARAMETER

ITL - WILL CONTAIN ONE OF:

THE CURRENT ETL VALUE -OR-

0 (THE DEFAULT VALUE - SEE REMARKS)

CM REQUIRED: 43B

SUBPROGRAMS REQUIRED

PART OF LANGUAGE

AND MASK

OTHERS

RFJTBL - READ INTERCOM USER TABLE

ARITHMETIC STATEMENT FUNCTIONS

GETBITS - EXTRACT BITS FROM A WORD

AUTHOR

DAVID V. SOMMER - DTNSRDC CODE 1892.2

DATE WRITTEN: 10/05/84

DATE(S) REVISED

LOCATION OF DECKS

SOURCE DECK

UPDATE LIBRARY ON MSS: NSRDCPL,UN=CSYS

OBJECT DECK

EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC,,ITL,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'JGDATE'

PURPOSE

CONVERT ANY GREGORIAN DATE TO A RELATIVE JULIAN DATE OR VICE
VERSA (FOR MULTI-YEAR COMPUTATIONS)

FUNCTIONAL CATEGORIES: M2

LANGUAGE: FORTRAN IV

COMPUTERS (OPERATING SYSTEMS)
CDC 6000/CYBER 170 (NOS/BE)

REMARKS

JG=1 IS VALID FOR ANY GREGORIAN DATE PRODUCING A RELATIVE
JULIAN DATE GREATER THAN ZERO.

THIS SUBROUTINE IS USEFUL IN DETERMINING THE ELAPSED NUMBER
OF DAYS BETWEEN ANY TWO CALENDAR DATES. IT CAN ALSO BE USED
TO FIND THE CALENDAR DATE SO MANY DAYS FROM ANY GIVEN DATE.

THE RELATIVE JULIAN DATE CORRESPONDING TO A GREGORIAN DATE
HAS MEANING TO THIS SUBROUTINE ONLY. IT REPRESENTS THE
NUMBER OF DAYS SINCE 11/24/-4713 (EXTRAPOLATING THE
GREGORIAN CALENDAR).

SEE ALSO SUBROUTINE 'JULIAN' FOR DAY-OF-YEAR DETERMINATION.

USAGE

CALL JGDATE (JG, JD, IGY, IGM, IGD)

DESCRIPTION OF PARAMETERS

JG - DIRECTION OF CONVERSION
1 - GREGORIAN TO RELATIVE JULIAN
2 - RELATIVE JULIAN TO GREGORIAN
JD - RELATIVE JULIAN DATE (OUT IF JG=1, IN IF JG=2)
IGY - GREGORIAN YEAR (EG, 1975) (IN IF JG=1, OUT IF JG=2)
IGM - GREGORIAN MONTH (1-12) (IN IF JG=1, OUT IF JG=2)
IGD - GREGORIAN DAY (1-31) (IN IF JG=1, OUT IF JG=2)

CM REQUIRED: 72B

SUBPROGRAMS REQUIRED
PART OF LANGUAGE
NONE
OTHERS
NONE

ARITHMETIC STATEMENT FUNCTIONS
NONE

METHOD
SEE COMM. OF THE ACM, VOL. 11, NO. 10, OCT 1968, PAGE 657.

AUTHOR
?

DATE WRITTEN: 1968 OR EARLIER

DATE(S) REVISED
03/01/79 - IMPLEMENT ON B7700
03/21/83 - RECOMPILE AT NOS/BE LEVEL 552

LOCATION OF DECKS
SOURCE
UPDATE LIBRARY ON MSS: NSRDCPL,UN=CSYS
OBJECT
EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT
BEGIN,DOCGET,,NSRDC,,JGDATE,OUTPUT,MSACCES=<PASSWORD>.

FUNCTION 'JOBCEM'

PURPOSE

GET JOB CARD CE

FUNCTIONAL CATEGORIES: QO

LANGUAGE: FORTRAN IV EXTENDED

COMPUTERS (OPERATING SYSTEMS)

CDC 6000/CYBER 170 (NOS/BE)

REMARKS

NONE

USAGE

JCE = JOBCM (JBCE)

CALL JOBCM (JBCE)

DESCRIPTION OF PARAMETERS

JOBCM - WILL CONTAIN JOB CARD CE

JBCE - WILL ALSO CONTAIN JOB CARD CE

CE REQUIRED: 25B

SUBPROGRAMS REQUIRED

PART OF LANGUAGE

AND SHIFT

OTHERS

RCEA - READ CONTROL POINT AREA

ARITHMETIC STATEMENT FUNCTIONS

R27FMT - FAST R-FORMAT DECODE (RIGHT-ADJ, ZERO-FILLED)

METHOD

THE JOB CARD CE IS TAKEN FROM CONTROL POINT AREA.

AUTHOR

DAVID V SOMMER - DTNSRDC CODE 1892.2

DATE WRITTEN: 10/16/81

DATE(S) REVISED

03/23/83 - RECOMPILE AT NOS/BE LEVEL 552

LOCATION OF DECKS

SOURCE

UPDATE LIBRARY ON MSS: NSRDCPL,UN=CSYS

OBJECT

EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC,,JOBCE,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'JOBNAME'
FUNCTION 'JOBNAME'

PURPOSE

GET SYSTEM JOB NAME FOR THIS JOB

FUNCTIONAL CATEGORIES: Q0

LANGUAGE: FORTRAN IV

COMPUTERS (OPERATING SYSTEMS)
CDC 6000/CYBER 170 (NOS/BE)

REMARKS

NONE

USAGE

CALL JOBNAME (I)
VARIABLE = JOBNAME (I)

DESCRIPTION OF PARAMETERS

JOBNAME - WILL CONTAIN JOB NAME, LEFT-JUSTIFIED,
ZERO-FILLED (WHEN USED AS FUNCTION)
I - WILL CONTAIN JOB NAME, LEFT-JUSTIFIED,
ZERO-FILLED

CM REQUIRED: 25B

METHOD

THE JOB NAME IS TAKEN FROM THE FIRST 7 CHARACTERS OF
CONTROL POINT AREA + 25B

SUBPROGRAMS REQUIRED

PART OF LANGUAGE
NONE

OTHERS

RCPA - READ CONTROL POINT AREA

AUTHOR

DAVID V SOMMER - DTNSRDC CODE 1892.2

DATE WRITTEN: 12/04/75

DATE(S) REVISED

03/23/83 - RECOMPILE AT NOS/BE LEVEL 552

LOCATION OF DECKS

SOURCE

UPDATE LIBRARY ON MSS: NSRDCPL,UN=CSYS

OBJECT

EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC,,JOBNAME,OUTPUT,MSACCES=<PASSWORD>.

FUNCTION 'JOBORG'
SUBROUTINE 'JOBORG'

PURPOSE
DETERMINE JOB ORIGIN

FUNCTIONAL CATEGORIES: QO

LANGUAGE: FORTRAN IV EXTENDED

COMPUTERS (OPERATING SYSTEMS)
CDC 6000/CYBER 170 (NOS/BE)

REMARKS
NONE

USAGE
IVAR = JOBORG (I, IA)
IVAR = JOBORG (I)
CALL JOBORG (I, IA)
CALL JOBORG (I)

DESCRIPTION OF PARAMETERS

I - WILL CONTAIN ONE OF THE FOLLOWING:

- 1 - IF CALLING JOB IS A BATCH JOB
- 2 - FOR REAL TIME JOB
- 3 - FOR GRAPHICS JOB
- 4 - FOR MULTI-USER JOB
- 5 - FOR INTERCOM

IA - IF SPECIFIED, WILL CONTAIN: 'BATCH', 'REAL TIME',
'GRAPHICS', 'MULTI-USER', OR 'INTERCOM', ACCORDING
TO THE VALUE OF 'I'.

IF USED AS A FUNCTION, 'JOBORG' WILL RETURN THE SAME VALUE
AS 'I'.

CM REQUIRED: 35B

METHOD
THE INFORMATION IS TAKEN FROM THE CONTROL POINT AREA.

SUBPROGRAMS REQUIRED
PART OF LANGUAGE
AND
OTHERS
RCPA - READ CONTROL POINT AREA

AUTHOR

DAVID V SOMMER - DTNSRDC CODE 1892.2

DATE WRITTEN: 03/07/77

DATE(S) REVISED

03/23/83 - RECOMPILE AT NOS/BE LEVEL 552

LOCATION OF DECKS

SOURCE

UPDATE LIBRARY ON MSS: NSRDCPL,UN=CSYS

OBJECT

EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC,,JOBORG,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'JULIAN'

PURPOSE

CONVERT ANY GREGORIAN DATE TO A JULIAN DAY-OF-YEAR OR VICE
VERSA (FOR SINGLE YEAR COMPUTATIONS ONLY)

FUNCTIONAL CATEGORIES: M2

LANGUAGE: FORTRAN IV

COMPUTERS (OPERATING SYSTEMS)
CDC 6000/CYBER 170 (NOS/BE)

REMARKS

THE PARAMETER 'IGY' IS ALWAYS INPUT.

IF JG=1 AND (GM<1 OR GM>12 OR GD<1 OR GD>31),
THEN JD IS SET TO ZERO (0).

IF JG=2 AND (JD<1 OR JD>366), THEN GM IS SET TO ZERO (0).

IF JG IS NOT 1 OR 2, THEN JD AND GM ARE SET TO ZERO (0).

SEE ALSO SUBROUTINE 'JGDATE' FOR MULTI-YEAR COMPUTATIONS.

USAGE

CALL JULIAN (JG, JD, IGY, IGM, IGD)

DESCRIPTION OF PARAMETERS

JG - DIRECTION OF CONVERSION
1 - GREGORIAN TO JULIAN
2 - JULIAN TO GREGORIAN
JD - JULIAN DAY-OF-YEAR (1-366)
IGY - GREGORIAN YEAR (E.G., 1968, ALWAYS INPUT)
IGM - GREGORIAN MONTH (1-12)
IGD - GREGORIAN DAY (1-31)

CM REQUIRED: 141B

REFERENCE

COMMUNICATIONS OF THE ACM, VOL II, NO 10, OCTOBER 1968, PAGE
657.

SUBPROGRAMS REQUIRED
PART OF LANGUAGE
MOD
OTHERS
NONE

AUTHOR
DAVID V SOMMER - DTNSRDC CODE 1892.2

DATE WRITTEN: 1968

DATE(S) REVISED
04/26/73 - REWRITTEN IN FORTRAN FOR CDC 6000 - DVS
06/21/76
01/11/78
03/01/79 - IMPLEMENTED ON B7700
03/23/83 - RECOMPILE AT NOS/BE LEVEL 552

LOCATION OF DECKS
SOURCE
UPDATE LIBRARY ON MSS: NSRDCPL,UN=CSYS
OBJECT
EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT
BEGIN,DOCGET,,NSRDC,,JULIAN,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'KUTMER'

PURPOSE

INTEGRATE A SYSTEM OF FIRST-ORDER ORDINARY DIFFERENTIAL EQUATIONS USING THE KUTTA-MERSON FOURTH-ORDER, SINGLE-STEP METHOD

FUNCTIONAL CATEGORIES: D2

LANGUAGE: FORTRAN IV

COMPUTERS (OPERATING SYSTEMS)
CDC 6000/CYBER 170 (NOS/BE)

REMARKS

THIS ROUTINE WILL INTEGRATE A SYSTEM OF FIRST-ORDER DIFFERENTIAL EQUATIONS OF THE FORM

$$\frac{d\bar{Y}}{dt} = \bar{F}(T, \bar{Y})$$

GIVEN A SET OF INITIAL CONDITIONS $T_0, \bar{Y}(T_0),$

AN INTERVAL H AND A SUBROUTINE FOR EVALUATING F-BAR(T, Y-BAR) FOR SPECIFIED VALUES OF T AND Y-BAR.

THE DIMENSIONS OF THE ARRAYS FOR STORING INTERMEDIATE VALUES OF THE VECTORS F-BAR AND Y-BAR ARE PRESENTLY SET TO 10. THIS CAN BE READILY CHANGED BY CHANGING THE DIMENSION STATEMENT AT THE BEGINNING OF THE SUBROUTINE.

USAGE

CALL KUTMER (N, T, Y, EPS, H, FIRST, HCX, A)

DESCRIPTION OF PARAMETERS

- N - NUMBER OF EQUATIONS (I.E., THE NUMBER OF COMPONENTS IN Y-BAR)
- T - THE INDEPENDENT VARIABLE, T
- Y - THE ARRAY OF DEPENDENT VARIABLES, Y-BAR
- EPS - THE RELATIVE ERROR CRITERION FOR EACH STEP, TO BE USED FOR THOSE COMPONENTS OF Y-BAR WHICH ARE GREATER THAN A IN ABSOLUTE VALUE
- H - THE INTEGRATION INTERVAL, H
- FIRST - WILL HAVE ONE OF THE FOLLOWING SETTINGS:
 - 0 - WHEN KUTMER IS ENTERED FOR THE FIRST TIME, OR IS RE-ENTERED WITH A CHANGED INTERVAL <H>. WHEN KUTMER IS SO ENTERED, <FIRST> IS RESET BY KUTMER TO 1.
 - 1 - WHEN KUTMER IS RE-ENTERED WITH THE SAME INTERVAL <H>, TO CONTINUE AN INTEGRATION SEQUENCE. UNDER THESE CIRCUMSTANCES, KUTMER WILL NOT RESET <FIRST>.

- 2 - WHEN KUTMER CANNOT MEET THE SPECIFIED ERROR CRITERIA EVEN WHEN THE INTEGRATION STEP HAS BEEN REDUCED TO $H/128$. KUTMER WILL RESET $\langle \text{FIRST} \rangle$ TO 2 AND PRINT A STATEMENT INDICATING THAT THE ERROR CRITERION COULD NOT BE MET. THEN KUTMER WILL RETURN CONTROL TO THE CALLING PROGRAM.
- HGX - IS SET UP BY KUTMER BEFORE EACH RETURN TO THE CALLING PROGRAM. THIS WILL CONTAIN THE MINIMUM STEP SIZE USED DURING THE INTEGRATION OVER THE INTERVAL $\langle H \rangle$.
- A - AN ABSOLUTE ERROR CRITERION TO BE USED FOR ANY COMPONENT OF $Y\text{-BAR}$ WHENEVER IT BECOMES SMALLER IN ABSOLUTE VALUE THAN $\langle A \rangle$.

ON ENTRY, $\langle T \rangle$ AND THE ARRAY $\langle Y \rangle$ CONTAIN VALUES OF THE INDEPENDENT AND THE DEPENDENT VARIABLES, RESPECTIVELY, AT THE BEGINNING OF THE INTERVAL OF INTEGRATION. ON RETURN, PROVIDED THE ERROR CRITERION HAS BEEN MET, I.E., $\langle \text{FIRST} \rangle$ HAS NOT BEEN RESET TO 2, $\langle T \rangle$ AND $\langle Y \rangle$ CONTAIN VALUES OF T AND $Y\text{-BAR}$ AT THE END VALUES OF THE INTEGRATION INTERVAL OF $\langle H \rangle$.

A SUBROUTINE FOR EVALUATING $F\text{-BAR}(T, Y\text{-BAR})$ WITH A CALL OF THE FORM

CALL DAUX (T, Y, F)
MUST BE SUPPLIED. HERE $\langle T \rangle$ AND THE ARRAY $\langle Y \rangle$ REFER TO T AND $Y\text{-BAR}$, RESPECTIVELY, AND THE ARRAY $\langle F \rangle$ SHOULD CONTAIN, ON RETURN FROM THIS SUBROUTINE, THE VECTOR $F\text{-BAR}(T, Y\text{-BAR})$.

CM REQUIRED: 352B

OUTPUT UNITS

UNIT #	LFN	USE
-----	-----	-----
	OUTPUT	ERROR MESSAGE

METHOD

THE KUTTA-MERSON METHOD OF INTEGRATING A SYSTEM OF FIRST-ORDER ORDINARY DIFFERENTIAL EQUATIONS IS USED. THIS IS A FOURTH-ORDER, SINGLE-STEP METHOD WHICH PROVIDES A CONVENIENT TECHNIQUE FOR AUTOMATIC INTERVAL ADJUSTMENT (C.F., E. FOX, "NUMERICAL SOLUTION OF ORDINARY AND PARTIAL DIFFERENTIAL EQUATIONS", ADDISON-WESLEY, READING, MASS., 1962, P. 24). THE ROUTINE IS BASICALLY A TRANSLATION INTO FORTRAN OF ALGOL ALGORITHM 218 PUBLISHED IN "COMMUNICATIONS OF THE ACM", DEC. 1963.

SUBPROGRAMS REQUIRED

PART OF LANGUAGE

ABS

OTHERS

DAUX - USER-SUPPLIED SUBROUTINE TO EVALUATE F-BAR

AUTHOR

E. CUTHILL - DTNSRDC CODE 1805

DATE WRITTEN: 10/29/64 (FORTRAN VERSION)

DATE(S) REVISED

09/20/83 - RECOMPILE AT NOS/BE LEVEL 552

LOCATION OF DECKS

SOURCE

UPDATE LIBRARY ON MSS: NSRDCPM,UN=CSYS

OBJECT

EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC,,KUTMER,OUTPUT,MSACCES=<PASSWORD>.

FUNCTION 'LASTC'

PURPOSE

DETERMINE NUMBER OF CHARACTERS THRU LAST NON-BLANK
(NON-ZERO (OOB))

FUNCTIONAL CATEGORIES: M5

LANGUAGE: FORTRAN IV EXTENDED

COMPUTERS (OPERATING SYSTEMS)
CDC 6000/CYBER 170 (NOS/BE)

REMARKS

THE WORD IN 'A' WHICH CONTAINING THE LAST NON-BLANK (NON-
ZERO) CHARACTER IS (LASTC(A,N)+9)/10

USAGE

LASTC (A)
LASTC (A, N)

DESCRIPTION OF PARAMETERS

A - ARRAY TO BE SCANNED
N - NUMBER OF WORDS IN 'A' TO BE PROCESSED
LASTC - WILL CONTAIN THE NUMBER OF CHARACTERS IN 'A'
EXCLUDING TRAILING BLANKS (ZEROS)

CM REQUIRED: 37B

SUBPROGRAMS REQUIRED

PART OF LANGUAGE
LOCF SHIFT
OTHERS
NONE

AUTHOR

DAVID V SOMMER - DTNSRDC CODE 1892.2

DATE WRITTEN: 01/06/76

DATE(S) REVISED

07/25/77 - MAKE PARAMETER 'N' OPTIONAL
02/15/83 - RECOMPILE AT NOS/BE LEVEL 552

LOCATION OF DECKS

SOURCE
UPDATE LIBRARY ON MSS: NSRDCPL,UN=CSYS
OBJECT
EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC,,LASTC,OUTPUT,MSACCES=<PASSWORD>.

FUNCTION 'LASTCH'

PURPOSE

DETERMINE NUMBER OF CHARACTERS THRU LAST NON-BLANK

FUNCTIONAL CATEGORIES: M5

LANGUAGE: FORTRAN IV EXTENDED

COMPUTERS (OPERATING SYSTEMS)

CDC 6000/CYBER 170 (NOS/BE)

REMARKS

THE WORD IN 'A' WHICH CONTAINS THE LAST NON-BLANK CHARACTER
IS (LASTCH(A,N)+9)/10.

USAGE

LASTCH (A, NCHAR)

DESCRIPTION OF PARAMETERS

A - ARRAY TO BE SCANNED

NCHAR - NUMBER OF CHARACTERS IN 'A' TO BE PROCESSED

LASTCH - WILL CONTAIN THE NUMBER OF CHARACTERS IN 'A'
EXCLUDING TRAILING BLANKS

CM REQUIRED: 71B

SUBPROGRAMS REQUIRED

PART OF LANGUAGE

AND MOD

OTHERS

NONE

AUTHOR

DAVID V SOMMER - DTNSRDC CODE 1892.2

DATE WRITTEN: 02/13/79

DATE(S) REVISED

03/23/83 - RECOMPILE AT NOS/BE LEVEL 552

LOCATION OF DECKS

SOURCE

UPDATE LIBRARY ON MSS: NSRDCPL,UN=CSYS

OBJECT

EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC,,LASTCH,OUTPUT,MSACCES=<PASSWORD>.

FUNCTION 'LASTWRD'

PURPOSE

DETERMINE SUBSCRIPT OF LAST WORD OF ARRAY WHICH CONTAINS A
NON-BLANK

FUNCTIONAL CATEGORIES: M5

LANGUAGE: FORTRAN IV

COMPUTERS (OPERATING SYSTEMS)
CDC 6000/CYBER 170 (NOS/BE)

REMARKS
NONE

USAGE
LASTWRD (A, N)

DESCRIPTION OF PARAMETERS

LASTWRD - WILL CONTAIN SUBSCRIPT OF LAST WORD OF ARRAY WHICH
CONTAINS A NON-BLANK (AND NON-OOB)

A - ARRAY TO BE SCANNED

N - NUMBER OF WORDS IN 'A' TO BE PROCESSED

CM REQUIRED: 22B

SUBPROGRAMS REQUIRED

PART OF LANGUAGE
NONE

OTHERS
LASTC - FIND LAST NON-BLANK/NON-OOB CHARACTER IN ARRAY

AUTHOR
DAVID V SOMMER - DTNSRDC CODE 1892.2

DATE WRITTEN: 03/15/76

DATE(S) REVISED
03/23/83 - RECOMPILE AT NOS/BE LEVEL 552

LOCATION OF DECKS

SOURCE
UPDATE LIBRARY ON MSS: NSRDCPL,UN=CSYS
OBJECT
EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT
BEGIN.DOCGET,,NSRDC,,LASTWRD,OUTPUT.MSACCES=<PASSWORD>.

AD-A148 792

COMPUTER CENTER CDC LIBRARIES/NSRD (SUBPROGRAMS)(U)
DAVID W TAYLOR NAVAL SHIP RESEARCH AND DEVELOPMENT
CENTER BET. D V SOMMER ET AL. JUN 84

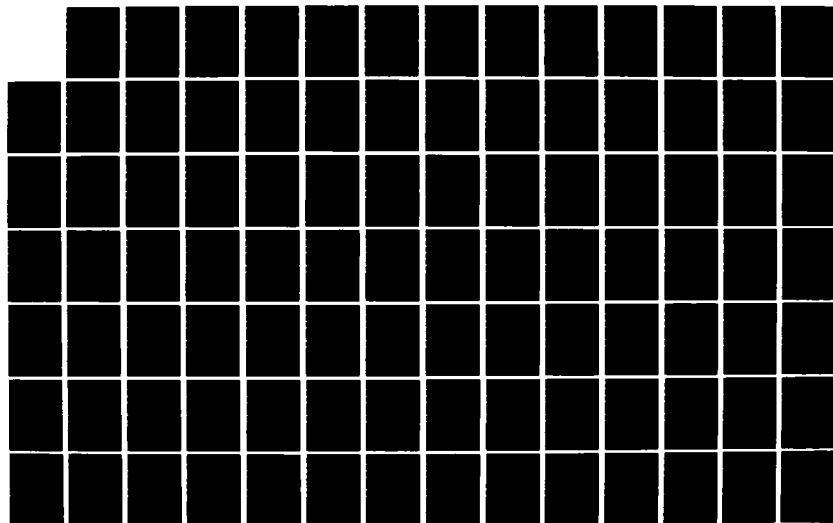
3/5

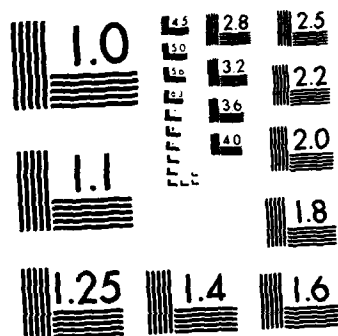
UNCLASSIFIED

DTNSRDC/CMLD-84-12

F/G 9/2

NL





MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A

FUNCTION 'LBYT'

PURPOSE

EXTRACT VARIABLE LENGTH BYTE

FUNCTIONAL CATEGORIES: M4

LANGUAGE: CDC 6000 CP COMPASS

COMPUTERS (OPERATING SYSTEMS)

CDC 6000/CYBER 170 (NOS/BE)

REMARKS

EXTRACTS A BYTE OF ANY LENGTH (1-60 BITS) FROM A 60-BIT WORD. THE EXTRACTED BYTE IS THEN STORED RIGHT-JUSTIFIED INTO ANOTHER 60-BIT WORD.

USAGE

VARIABLE = LBYT (N, LENGTH, FROM)

DESCRIPTION OF PARAMETERS

VARIABLE - LOCATION INTO WHICH THE EXTRACTED BYTE IS STORED RIGHT-JUSTIFIED

N - STARTING BIT POSITION OF THE BYTE TO BE EXTRACTED. BITS ARE NUMBERED 1-60 FROM RIGHT TO LEFT.

LENGTH - LENGTH OF THE BYTE (NUMBER OF BITS)

FROM - WORD FROM WHICH THE BYTE IS TO BE EXTRACTED

CM REQUIRED: 16B

EXAMPLE

STARTING AT THE TWELFTH BIT FROM THE RIGHT OF A WORD, A FOUR-BIT BYTE WILL BE EXTRACTED FROM THE VARIABLE <TAKE> AND STORED IN VARIABLE <ISTORE> IN BIT PLACES 1-4.

TAKE = 1111 2222 3333 4476 5555B

ISTORE = LBYT (12, 4, TAKE)

RESULTS IN

ISTORE = 0000 0000 0000 0000 0016B

NOTE: BIT POSITIONS 12-15 OF <TAKE> ARE 1 1 1 0.

SUBPROGRAMS REQUIRED
PART OF LANGUAGE
NONE
OTHERS
NONE

AUTHOR
FROM CDC KRONOS SYSTEM

DATE WRITTEN:

DATE(S) REVISED
03/22/83 - RECOMPILE AT NOS/BE LEVEL 552

LOCATION OF DECKS
SOURCE
UPDATE LIBRARY ON MSS: NSRDCPL,UN=CSYS
OBJECT
EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT
BEGIN,DOCGET,,NSRDC,,LBYT,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'LEFTADJ'

PURPOSE

SQUEEZE LEFT AND REMOVE BLANKS AND OOB (USER MAY SUPPLY
TRAILING FILL CHARACTER)

FUNCTIONAL CATEGORIES: M4

LANGUAGE: FORTRAN IV EXTENDED

COMPUTERS (OPERATING SYSTEMS)
CDC 6000/CYBER 170 (NOS/BE)

REMARKS

THE LAST NON-BLANK CHARACTER POSITION AND WORD ARE RETURNED.

USAGE

CALL LEFTADJ (A, NA, LASTC, NW, FILL)
CALL LEFTADJ (A, NA, LASTC, NW)

DESCRIPTION OF PARAMETERS

A - ARRAY TO BE LEFT JUSTIFIED
NA - NUMBER OF WORDS IN 'A' TO BE PROCESSED
LASTC - WILL RETURN THE LAST CHARACTER POSITION
WHICH IS NON-BLANK/NON-OOB (LEFT-MOST CHARACTER
POSITION IS 1)
(IF ARRAY CONTAINS ONLY BLANKS AND/OR OOB, LASTC IS
SET TO 0)
NW - WILL RETURN SUBSCRIPT OF WORD CONTAINING LAST
NON-BLANK/NON-OOB CHARACTER
(IF LASTC=0, THEN NW IS SET TO 0)
FILL - OPTIONAL FILL CHARACTER FOR EACH CHARACTER
POSITION AFTER LASTC (USE 1R OR 1H FORMAT)
(IF OMITTED, FILL CHARACTER IS OOB)

CM REQUIRED: 100B

EXAMPLE

DIMENSION A(4)
CONTENTS OF A: 12345 67890 ABCDEFGHIJ
CALL LEFTADJ (A, 4, LASTC, NW)
CONTENTS OF A: 1234567890ABCDEFGHIJ
LASTC IS 20; NW = 2
CALL LEFTADJ (A, 4, LASTC, NW, 1R/)
CONTENTS OF A: 1234567890ABCDEFGHIJ/////////////////
LASTC AND NW ARE THE SAME

SUBPROGRAMS REQUIRED

PART OF LANGUAGE

LOCF

OTHERS

GETCHA - EXTRACT ONE CHARACTER FROM AN ARRAY

PUTCHA - INSERT ONE CHARACTER INTO AN ARRAY

ARITHMETIC STATEMENT FUNCTIONS

R11FMT - FAST R-FORMAT DECODE (RIGHT-ADJ, ZERO-FILLED)

AUTHOR

DAVID V SOMMER - DTNSRDC CODE 1892.2

DATE WRITTEN: 11/02/76

DATE(S) REVISED

03/23/83 - RECOMPILE AT NOS/BE LEVEL 552

LOCATION OF DECKS

SOURCE

UPDATE LIBRARY ON MSS: NSRDCPL,UN-CSYS

OBJECT

EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC,,LEFTADJ,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'LPPFERR'

PURPOSE

DECODE THE "ERR" CODE FROM FILE MANIPULATION SUBROUTINES PF
AND LF

FUNCTIONAL CATEGORIES: Q0

LANGUAGE: FORTRAN IV EXTENDED

COMPUTERS (OPERATING SYSTEMS)
CDC 6000/CYBER 170 (NOS/BE)

REMARKS

NONE

USAGE

CALL LPPFERR (ERR, PARMNO, CODE)

DESCRIPTION OF PARAMETERS

ERR - "ERR" RETURNED BY PF OR LF
PARMNO - PARAMETER NUMBER
CODE - ERROR CODE NUMBER

CM REQUIRED: 13B

METHOD

THE PARAMETER NUMBER IS EXTRACTED FROM BITS 35-19 OF ERR;
THE ERROR CODE NUMBER IS EXTRACTED FROM BITS 18-0.

SUBPROGRAMS REQUIRED

PART OF LANGUAGE
AND SHIFT
OTHERS
NONE

ARITHMETIC STATEMENT FUNCTIONS

FAST R-FORMAT DECODE (RIGHT-ADJUSTED, ZERO-FILLED)
R35FMT R38FMT

AUTHOR

DAVID V SOMMER - DTNSRDC CODE 1892.2

DATE WRITTEN: 02/24/83

DATE(S) REVISED

03/23/83 - RECOMPILE AT NOS/BE LEVEL 552

LOCATION OF DECKS

SOURCE DECK
UPDATE LIBRARY ON MSS: NSRDCPL,UN=CSYS
OBJECT DECK
EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC,,LPPFERR,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'LIBBAM'

PURPOSE

DUMMY SUBROUTINE TO FORCE LDSET,LIB=BAMLIB

FUNCTIONAL CATEGORIES: QO

LANGUAGE: CDC 6000 CP COMPASS

COMPUTERS (OPERATING SYSTEMS)

CDC 6000/CYBER 170 (NOS/BE)

REMARKS

THIS SUBROUTINE IS USED TO FORCE A LOADER TABLE ENTRY FOR 'LDSET,LIB=BAMLIB' TO BE GENERATED SO THAT THE USER DOES NOT HAVE TO SUPPLY A LOADER DIRECTIVE EACH TIME. THIS IS NEEDED ONLY IF ROUTINES ON BAMLIB ARE USED AND NOTHING IN YOUR PROGRAM FORCES THE COMPILER TO GENERATE THIS LOADER TABLE ENTRY. AN EXAMPLE IS SUBROUTINE 'UNLOAD' TO UNLOAD A FILE. IT LOCATES THE FORTRAN FILE IN THE PROGRAM'S LIST OF FILES AND CALLS RECORD MANAGER ROUTINE 'CLOSEM' TO UNLOAD THE FILE. IF THIS IS THE ONLY RECORD MANAGER CALL IN THE PROGRAM, NO REFERENCE TO BAMLIB IS GENERATED BY THE COMPILER OR AN OBJECT-TIME SUBPROGRAM. 'UNLOAD', THEREFORE, HAS A DUMMY 'CALL LIBBAM', WHICH IS NEVER EXECUTED, THOUGH IT COULD BE.

USAGE

CALL LIBBAM

CM REQUIRED: 2

SUBPROGRAMS REQUIRED

PART OF LANGUAGE

NONE

OTHERS

NONE

AUTHOR

DAVID V SOMMER - DTNSRDC CODE 1892.2

DATE WRITTEN: 07/14/81

DATE(S) REVISED

03/22/83 - RECOMPILE AT NOS/BE LEVEL 552

LOCATION OF DECKS

SOURCE

UPDATE LIBRARY ON MSS: NSRDCPL,UN-CSYS

OBJECT

EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC,,LIBBAM,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'LIBSYM'

PURPOSE

DUMMY SUBROUTINE TO FORCE LDSET,LIB=SYMLIB

FUNCTIONAL CATEGORIES: Q0

LANGUAGE: CDC 6000 CP COMPASS

COMPUTERS (OPERATING SYSTEMS)

CDC 6000/CYBER 170 (NOS/BE)

REMARKS

THIS SUBROUTINE IS USED TO FORCE A LOADER TABLE ENTRY FOR 'LDSET,LIB=SYMLIB' TO BE GENERATED SO THAT THE USER DOES NOT HAVE TO SUPPLY A LOADER DIRECTIVE EACH TIME. THIS IS NEEDED ONLY IF FORTRAN-CALLABLE CMM ROUTINES ARE USED AND NOTHING IN YOUR PROGRAM FORCES THE COMPILER TO GENERATE THIS LOADER TABLE ENTRY.

USAGE

CALL LIBSYM

CM REQUIRED: 2B

SUBPROGRAMS REQUIRED

PART OF LANGUAGE

NONE

OTHERS

NONE

AUTHOR

DAVID V SOMMER - DTNSRDC CODE 1892.2

DATE WRITTEN: 02/15/83

DATE(S) REVISED

03/22/83 - RECOMPILE AT NOS/BE LEVEL 552

LOCATION OF DECKS

SOURCE

UPDATE LIBRARY ON MSS: NSRDCPL,UN=CSYS

OBJECT

EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC,,LIBSYM,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'LINE6'

PURPOSE

SET PRINT FILE TO 6 LINES PER INCH

FUNCTIONAL CATEGORIES: J4

LANGUAGE: FORTRAN IV

COMPUTERS (OPERATING SYSTEMS)

CDC 6000/CYBER 170 (NOS/BE)

REMARKS

YOU SHOULD PRINT YOUR NEXT LINE AT THE TOP OF THE NEXT PAGE.

USAGE

CALL LINE6 (IOUT)

DESCRIPTION OF PARAMETER

IOUT - OUTPUT UNIT NUMBER (1-99) OR NAME (1-7 CHARACTERS,
LEFT-JUSTIFIED, ZERO-FILLED)

CM REQUIRED: 20B

OUTPUT UNIT

UNIT #	LFN	USE
IOUT		LISTABLE OUTPUT FILE

SUBPROGRAMS REQUIRED

PART OF LANGUAGE

NONE

OTHERS

NONE

AUTHOR

DAVID V SOMMER - DTNSRDC CODE 1892.2

DATE WRITTEN: 06/11/76

DATE(S) REVISED

03/23/83 - RECOMPILE AT NOS/BE LEVEL 552

LOCATION OF DECKS

SOURCE

UPDATE LIBRARY ON MSS: NSRDCPL,UN=CSYS

OBJECT

EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC,,LINE6,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'LINE8'

PURPOSE

SET PRINT FILE TO 8 LINES PER INCH

FUNCTIONAL CATEGORIES: J4

LANGUAGE: FORTRAN IV

COMPUTERS (OPERATING SYSTEMS)

CDC 6000/CYBER 170 (NOS/BE)

REMARKS

YOU SHOULD PRINT YOUR NEXT LINE AT THE TOP OF THE NEXT PAGE.

USAGE

CALL LINE8 (IOUT)

DESCRIPTION OF PARAMETER

IOUT - OUTPUT UNIT NUMBER (1-99) OR NAME (1-7 CHARACTERS,
LEFT-JUSTIFIED, ZERO-FILLED)

CM REQUIRED: 20B

OUTPUT UNIT

UNIT #	LFN	USE
--------	-----	-----

IOUT		LISTABLE OUTPUT FILE
------	--	----------------------

SUBPROGRAMS REQUIRED

PART OF LANGUAGE

NONE

OTHERS

NONE

AUTHOR

DAVID V SOMMER - DTNSRDC CODE 1892.2

DATE WRITTEN: 06/11/76

DATE(S) REVISED

03/23/83 - RECOMPILE AT NOS/BE LEVEL 552

LOCATION OF DECKS

SOURCE

UPDATE LIBRARY ON MSS: NSRDCPL,UN=CSYS

OBJECT

EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC,,LINE8,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'MACHINE'

PURPOSE

RETURN 4-WORD SYSTEM HEADING

FUNCTIONAL CATEGORIES: Q0

LANGUAGE: CDC 6000 CP COMPASS

COMPUTERS (OPERATING SYSTEMS)

CDC 6000/CYBER 170 (NOS/BE)

REMARKS

MACHINE IS PART OF THE OPERATING SYSTEM, NOT IN LIBRARY
NSRDC. THE DOCUMENT IS HERE FOR CONVENIENCE.

USAGE

CALL MACHINE (ARRAY)

DESCRIPTION OF PARAMETER

ARRAY - 4-ELEMENT ARRAY WHICH WILL CONTAIN THE SYSTEM
HEADING

(E.G., ' NSRDC MFB NOS/BE 1.5 K+7 83013 ')

SUBPROGRAMS REQUIRED

PART OF LANGUAGE

NONE

OTHERS

NONE

CM REQUIRED: 25B

AUTHOR

DTNSRDC CODE 1892.3

DATE WRITTEN: 04/75

DATE(S) REVISED

LOCATION OF DECKS

SOURCE

CODE 1892.3

OBJECT

EDITLIB SYSTEM LIBRARY: SYSLIB

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC,,MACHINE,OUTPUT,MSACCES=<PASSWORD>.

FUNCTION 'MASKIT'

PURPOSE

DYNAMIC MASK GENERATOR

FUNCTIONAL CATEGORIES: MO

LANGUAGE: CDC 6000 CP COMPASS

COMPUTERS (OPERATING SYSTEMS)

CDC 6000/CYBER 170 (NOS/BE)

REMARKS

MASKIT GENERATES AS ITS FUNCTIONAL VALUE A WORD WITH 'N' FIELDS OF BITS SET, EACH FIELD 'FL' BITS LONG, AND STARTING AT BIT ADDRESS 'BIT'.

EXAMPLE: TO GENERATE THE MASK

11100011111111110111000000000100010000000111111000000111111
7 0 7 7 7 6 7 0 0 0 4 2 0 0 7 7 0 0 7 7

USE THE FOLLOWING:

MSK = MASKIT (3,59, 11,53, 3,41, 1,29, 1,25, 6,17, 6,5)

USAGE

MSK = MASKIT (FL1, BIT1, FL2, BIT2, ..., FLN, BITN)

DESCRIPTION OF PARAMETERS

FL - NUMBER OF BITS

BIT - STARTING BIT ADDRESS

BIT ADDRESSES ARE THE RELEVANT POWER OF 2.

I.E., 59,58,57,... ...,2,1,0

CM REQUIRED: 16B

SUBPROGRAMS REQUIRED

PART OF LANGUAGE

NONE

OTHERS

NONE

AUTHOR

C FLINK - KPS NWL

DATE WRITTEN: 07/70

DATE(S) REVISED

03/22/83 - RECOMPILE AT NOS/BE LEVEL 552

LOCATION OF DECKS

SOURCE

UPDATE LIBRARY ON MSS: NSRDCPL,UN=CSYS

OBJECT

EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC,,MASKIT,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'MATINS'

PURPOSE

MATRIX INVERSION WITH ACCOMPANYING SOLUTION OF SIMULTANEOUS
EQUATIONS AND DETERMINANT

FUNCTIONAL CATEGORIES: F4 F1 F3

LANGUAGE: FORTRAN IV

COMPUTERS (OPERATING SYSTEMS)
CDC 6000/CYBER 170 (NOS/BE)

REMARKS

TESTS FOR LOSS OF DIGITS DUE TO SUBTRACTION.

TO SCALE THE DETERMINANT, ROUTINE MUST BE RECOMPILED TO OMIT
INTERNAL 'DETERM = 1.'. IN THIS CASE, PARAMETER 'DETERM' IS
THE INPUT SCALING FACTOR AS WELL AS THE OUTPUT DETERMINANT.

USAGE

CALL MATINS (A, NR, N1, B, NC, M1, DETERM, ID, INDEX)

DESCRIPTION OF PARAMETERS

A - INPUT MATRIX (NR X NR)
(WILL BE REPLACED BY INVERSE OF 'A')
NR - REFERS TO CALLING PROGRAM DIMENSIONS:
ROWS IN 'A'; # COLUMNS IN 'A';
ROWS IN 'B'; # ROWS IN 'INDEX'
N1 - ORDER OF 'A'
(ACTUAL SIZE OF 'A' BEING USED)
B - COLUMN VECTORS
(WILL BE REPLACED BY CORRESPONDING SOLUTION
VECTORS)
NC - REFERS TO CALLING PROGRAM DIMENSIONS:
COLUMNS IN 'B'
M1 - NUMBER OF ACTUAL COLUMN VECTORS IN 'B'
(MAY BE 0)
DETERM - OUTPUT DETERMINANT
ID - OUTPUT CODE
1 - INVERSE SUCCESSFUL
2 - MATRIX 'A' SINGULAR
INDEX - WORKING STORAGE ARRAY OF DIMENSION (NR X 3)

NOTE: N1 <= NR; M1 <= NC

CM REQUIRED: 340B

METHOD

PIVOT METHOD - GAUSS-JORDAN

SUBPROGRAMS REQUIRED
PART OF LANGUAGE
ABS
OTHERS
NONE

AUTHORS
ANF402 FROM SHARE
SHARON E GOOD - DTNSRDC CODE 1892.1
C R NEWMAN - NOL

DATE WRITTEN: 11/71

DATE(S) REVISED
07/26/77 - ADD CRN CODING (SEG)
09/23/83 - RECOMPILE AT NOS/BE LEVEL 552

LOCATION OF DECKS
SOURCE
UPDATE LIBRARY ON MSS: NSRDCPM,UN=CSYS (*DECK AMMAT4)
OBJECT
EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT
BEGIN,DOCGET,,NSRDC,,MATINS,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'MAXE'
FUNCTION 'MAXE'
FUNCTION 'AMAXE'

PURPOSE

FIND MAXIMUM VALUE OF AN ARRAY

FUNCTIONAL CATEGORIES: M5

LANGUAGE: CDC 6000 CP COMPASS

COMPUTERS (OPERATING SYSTEMS)
CDC 6000/CYBER 170 (NOS/BE)

REMARKS

FUNCTION MAXE HAS INTEGER INPUT AND OUTPUT.
FUNCTION AMAXE HAS REAL INPUT AND OUTPUT.

USAGE

CALL MAXE (ARRAY, ISIZE, AMAXV)

MAXV = MAXE (IARRAY, ISIZE)
AMAXV = AMAXE (ARRAY, ISIZE)

DESCRIPTION OF PARAMETERS

ARRAY - REAL ARRAY TO BE PROCESSED
IARRAY - INTEGER ARRAY TO BE PROCESSED
ISIZE - LENGTH OF ARRAY/IARRAY
AMAXV - REAL MAXIMUM RETURNED IN SUBROUTINE

CM REQUIRED: 14B

REMARKS

FUNCTION MAXE HAS INTEGER INPUT AND OUTPUT.
FUNCTION AMAXE HAS REAL INPUT AND OUTPUT.

SUBPROGRAMS REQUIRED

PART OF LANGUAGE
NONE
OTHERS
NONE

AUTHOR

C FLINK - KPS NWL

DATE WRITTEN: 11/22/70

DATE(S) REVISED

03/23/83 - RECOMPILE AT NOS/BE LEVEL 552

LOCATION OF DECKS

SOURCE
UPDATE LIBRARY ON MSS: NSRDCPL,UN=CSYS
OBJECT
EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC,,MAXE,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'MEMUSED'

PURPOSE

PRINT MESSAGE IN DAYFILE GIVING FIELD LENGTH IN USE AT
TIME OF CALL TO THIS ROUTINE

FUNCTIONAL CATEGORIES: Q0

LANGUAGE: CDC 6000 CP COMPASS

COMPUTERS (OPERATING SYSTEMS)
CDC 6000/CYBER 170 (NOS/BE)

REMARKS

THIS ROUTINE ISSUES A MEMORY MACRO REQUEST TO DETERMINE
FIELD LENGTH AND PRINTS A MESSAGE IN THE DAYFILE OF
THE FORM:

FIELD LENGTH IN USE (OCTAL) = XXXXXX

IT MIGHT BE OF INTEREST TO USERS WITH PROGRAMS WHICH
MANAGE FIELD LENGTH DYNAMICALLY ABOVE THAT SHOWN IN THE
NORMAL LOAD MAP (SUCH AS FILE BUFFER SPACE IN COBOL
PROGRAMS).

USAGE

CALLED FROM COBOL PROGRAM
ENTER MEMUSED.

CALLED FROM FTN PROGRAM
CALL MEMUSED

CM REQUIRED: 30B

SUBPROGRAMS REQUIRED

PART OF LANGUAGE

NONE

OTHERS

NONE

AUTHOR

BRUCE D. BLACK - DTNSRDC CODE 1892.1 (CDC)

DATE WRITTEN: 04/07/78

DATE(S) REVISED

03/22/83 - RECOMPILE AT NOS/BE LEVEL 552

LOCATION OF DECKS

SOURCE

UPDATE LIBRARY ON MSS: NSRDCPL,UN=CSYS

OBJECT

EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC,,MEMUSED,OUTPUT,MSACCES=<PASSWORD>.

FUNCTION 'MFETCH'

PURPOSE

FETCH A SINGLE WORD (BY ABSOLUTE ADDRESS) FROM USER'S FL

FUNCTIONAL CATEGORIES: K2

LANGUAGE: CDC 6000 CP COMPASS

COMPUTERS (OPERATING SYSTEMS)

CDC 6000/CYBER 170 (NOS/BE)

REMARKS

'MFETCH' IS AN ENTRY POINT IN 'CMDRCT'.

USAGE

MFETCH (ADDR)

DESCRIPTION OF PARAMETER

ADDR - ADDRESS IN USER'S FL TO BE FETCHED

NO ERROR CHECKING IS DONE.

CM REQUIRED: 11B (INCLUDES 'MSET')

SUBPROGRAMS REQUIRED

PART OF LANGUAGE

NONE

OTHERS

NONE

AUTHOR

? - NWL

DATE WRITTEN:

DATE(S) REVISED

03/23/83 - RECOMPILE AT NOS/BE LEVEL 552

LOCATION OF DECKS

SOURCE

UPDATE LIBRARY ON MSS: NSRDCPL,UN=CSYS (*DECK CMDRCT)

OBJECT

EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC,,MFETCH,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'MFRAME'

PURPOSE

OBTAIN THE MACHINE AND MAINFRAME RUNNING THE PROGRAM

FUNCTIONAL CATEGORIES: Q0

LANGUAGE: FORTRAN IV EXTENDED

COMPUTERS (OPERATING SYSTEMS)
CDC 6000/CYBER 170 (NOS/BE)

REMARKS

NONE

USAGE

CALL MFRAME (CPU, MF)

DESCRIPTION OF PARAMETER

CPU - WILL RETURN MACHINE ON WHICH THE PROGRAM IS RUNNING
(CDC FTN4: INTEGER, LEFT-ADJ, BLANK-FILLED)
CDC FTN5: CHARACTER*6;
(WILL RETURN ONE OF:
"6700", "6600", "6400", "CY74", "CY750", "CY176",
"CY825")

MF - WILL RETURN MAINFRAME ON WHICH THE PROGRAM IS RUNNING
(CDC FTN4: INTEGER, LEFT-ADJ, BLANK-FILLED)
CDC FTN5: CHARACTER*3;
(WILL RETURN ONE OF:
"MFA", "MFB", "MFC", "MFD", "MFE", "MFF",
"MFG", "MFZ")

CM REQUIRED: 143B (FTN4); 320B (FTN5)

SUBPROGRAMS REQUIRED

PART OF LANGUAGE
AND SHIFT
OTHERS

MACHINE - GET SYSTEM MACHINE INFORMATION

ARITHMETIC STATEMENT FUNCTIONS

R38FMT - FAST R-FORMAT DECODE (RIGHT-ADJ, ZERO-FILLED)

AUTHOR

DAVID V SOMMER - DTNSRDC CODE 1892.2

DATE WRITTEN: 03/15/79

DATE(S) REVISED

08/15/80 - ADD "CY74" FOR CYBER 74
07/01/82 - UPGRADE TO LEVEL 552
- ADD SUPPORT FOR MFE, MFF, MFG
03/23/83 - RECOMPILE AT NOS/BE LEVEL 552
08/19/83 - CHANGE CPU FOR MFG

LOCATION OF DECKS

SOURCE

UPDATE LIBRARY ON MSS: NSRDCPL,UN=CSYS
NSRDC5P,UN=CSYS

OBJECT

EDITLIB USER LIBRARY: NSRDC AND NSRDC5

ANOTHER COPY OF THIS DOCUMENT

CDC FTN4: BEGIN,DOCGET,,NSRDC,,MFRAME,OUTPUT,MSACCES=<PW>.
CDC FTN5: BEGIN,DOCGET,,NSRDC5,,MFRAME,OUTPUT,MSACCES=<PW>.

FUNCTION 'MF2CPU'

PURPOSE

RETURN CPU NAME CORRESPONDING TO SUPPLIED MAINFRAME NAME

FUNCTIONAL CATEGORIES: Z0

LANGUAGE: FORTRAN IV

COMPUTERS (OPERATING SYSTEMS)

CDC 6000/CYBER 170 (NOS/BE)

REMARKS

NONE

USAGE

ICPU = MF2CPU (MF)

DESCRIPTION OF PARAMETERS

MF - INPUT MAINFRAME NAME OR LETTER
(E.G., "MFF" OR "F")

MF2CPU - WILL RETURN THE CORRESPONDING CPU NAME
(E.G., "CY176")

-OR- " ", IF MF IS NOT RECOGNIZED

CM REQUIRED: 111B

SUBPROGRAMS REQUIRED

PART OF LANGUAGE

NONE

OTHERS

NONE

AUTHOR

DAVID V SOMMER - DTNSRDC CODE 1892.2

DATE WRITTEN: 04/20/82

DATE(S) REVISED

03/23/83 - RECOMPILE AT NOS/BE LEVEL 552

LOCATION OF DECKS

SOURCE

UPDATE LIBRARY ON MSS: NSRDCPL,UN=CSYS

OBJECT

EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,NSRDC,MF2CPU,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'MINE'
FUNCTION 'MINE'
FUNCTION 'AMINE'

PURPOSE

FIND MINIMUM VALUE OF AN ARRAY

FUNCTIONAL CATEGORIES: M5

LANGUAGE: CDC 6000 CP COMPASS

COMPUTERS (OPERATING SYSTEMS)
CDC 6000/CYBER 170 (NOS/BE)

REMARKS

FUNCTION MINE HAS INTEGER INPUT AND OUTPUT.
FUNCTION AMINE HAS REAL INPUT AND OUTPUT.

USAGE

CALL MINE (ARRAY, ISIZE, AMINV)

MINV = MINE (IARRAY, ISIZE)
AMINV = AMINE (ARRAY, ISIZE)

DESCRIPTION OF PARAMETERS

ARRAY - REAL ARRAY TO BE PROCESSED
IARRAY - INTEGER ARRAY TO BE PROCESSED
ISIZE - LENGTH OF ARRAY/IARRAY
AMINV - REAL MINIMUM RETURNED IN SUBROUTINE

CM REQUIRED: 14B

SUBPROGRAMS REQUIRED

PART OF LANGUAGE
NONE
OTHERS
NONE

AUTHOR

C FLINK - KPS NWL

DATE WRITTEN: 11/22/70

DATE(S) REVISED

03/23/83 - RECOMPILE AT NOS/BE LEVEL 552

LOCATION OF DECKS

SOURCE
UPDATE LIBRARY ON MSS: NSRDCPL,UN=CSYS
OBJECT
EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC,,MINE,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'MONTH'

PURPOSE

FROM A DATE (MM/DD/YY) FIND THE MONTH AND RETURN FULL
SPELLING AND 3- OR 4-CHARACTER ABBREVIATION

FUNCTIONAL CATEGORIES: M2

LANGUAGE: FORTRAN IV

COMPUTERS (OPERATING SYSTEMS)
CDC 6000/CYBER 170 (NOS/BE)

REMARKS
NONE

USAGE

CALL MONTH (DATE, MONTH, MM)

DESCRIPTION OF PARAMETERS

DATE - DATE TO BE PROCESSED ('MM/DD/YY ', ' MM/DD/YY '
OR ' MM/DD/YY')
IMONTH - WILL CONTAIN THE MONTH (COMPLETE SPELLING)
MM - WILL CONTAIN THE MONTH (3- OR 4-CHARACTER
ABBREVIATION)

CM REQUIRED: 62B

SUBPROGRAMS REQUIRED

PART OF LANGUAGE
SHIFT
OTHERS
NONE

ARITHMETIC STATEMENT FUNCTIONS

I21FMT - FAST I-FORMAT DECODE
L11FMT - FAST L-FORMAT DECODE (LEFT-ADJ, ZERO-FILLED)

AUTHOR

DAVID V SOMMER - DTNSRDC CODE 1892.2

DATE WRITTEN: 07/21/76

DATE(S) REVISED

03/23/83 - RECOMPILE AT NOS/BE LEVEL 552

LOCATION OF DECKS

SOURCE
UPDATE LIBRARY ON MSS: NSRDCPL,UN=CSYS
OBJECT
EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC,,MONTH,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'MOVCHAR'

PURPOSE

MOVE ONE CHARACTER FROM ONE STRING TO ANOTHER

COMPUTERS (OPERATING SYSTEMS)

CDC 6000/CYBER 170 (NOS/BE)

FUNCTIONAL CATEGORIES: M4

LANGUAGE: FORTRAN IV EXTENDED

REMARKS

NONE

USAGE

CALL MOVCHAR (FROM, FROM COL, TO, TO COL)

DESCRIPTION OF PARAMETERS

FROM - ARRAY CONTAINING STRING FROM WHICH THE CHARACTER
IS TO BE EXTRACTED
FROM COL - POSITION OF CHARACTER IN FROM
(1 IS LEFTMOST POSITION)
TO - ARRAY TO WHICH THE CHARACTER IS TO BE MOVED
TO COL - POSITION OF CHARACTER IN TO
(1 IS LEFTMOST POSITION)

CM REQUIRED: 35B

EXAMPLE

BEFORE: FROM=THIS IS A CHARACTER STRING.
TO =THIS IS ANOTHER STRING
CALL MOVSTR (FROM, 27, TO, 23)
AFTER : TO =THIS IS ANOTHER STRING.

SUBPROGRAMS REQUIRED

PART OF LANGUAGE
AND MOD OR SHIFT
OTHERS
NONE

AUTHOR

DAVID V SOMMER - DTNSRDC CODE 1892.2

DATE WRITTEN: 11/14/77

DATE(S) REVISED

03/23/83 - RECOMPILE AT NOS/BE LEVEL 552

LOCATION OF DECKS

SOURCE
UPDATE LIBRARY ON MSS: NSRDCPL,UN-CSYS
OBJECT
EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC,,MOVCHAR,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'MOVECM'

PURPOSE

MOVE WORDS FROM ONE AREA IN CORE TO ANOTHER

FUNCTIONAL CATEGORIES: M4

LANGUAGE: CDC 6000 CP COMPASS

COMPUTERS (OPERATING SYSTEMS)

CDC 6000/CYBER 170 (NOS/BE)

REMARKS

'MOVECM' IS ABOUT 20 PERCENT FASTER THAN THE FTN-SUPPLIED 'MOVLEV'. IT MOVES 4 WORDS AT A TIME (INSTEAD OF 2) AND DOES NOT REQUIRE AT LEAST ONE CM WORD BETWEEN THE SENDING AND RECEIVING FIELDS.

AT SPEED, 'MOVECM' MOVES ABOUT 2 WORDS PER MICROSECOND.

USAGE

CALL MOVECM (FWA, LWA, NEW FWA)

DESCRIPTION OF PARAMETERS

FWA - FIRST WORD ADDRESS OF SENDING FIELD

LWA - LAST WORD ADDRESS OF SENDING FIELD

NEW FWA - FIRST WORD ADDRESS OF RECEIVING FIELD

(MOVE MEMORY WORDS BEGINNING AT FWA AND ENDING AT LWA TO A BLOCK STARTING AT NEW FWA.)

CM REQUIRED: 20B

EXAMPLE

MOVE ARRAY 'A' TO ARRAY 'B':

...

DIMENSION A(100), B(100)

...

CALL MOVECM (A(1), A(100), B(1))

...

METHOD

WORDS ARE MOVED 4 AT A TIME, UNLESS FEWER THAN 4 REMAIN TO BE MOVED.

SUBPROGRAMS REQUIRED
PART OF LANGUAGE
NONE
OTHERS
NONE

AUTHOR
EXTRACTED FROM 'NETED', THE TEXT EDITOR FROM ED FOURT OF
LAWRENCE BERKLEY LABS

DATE WRITTEN:

DATE(S) REVISED
03/23/83 - RECOMPILE AT NOS/BE LEVEL 552

LOCATION OF DECKS
SOURCE
UPDATE LIBRARY ON MSS: NSRDCPL,UN=CSYS
OBJECT
EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT
BEGIN,DOCGET,,NSRDC,,MOVECM,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'MOVEIT'

PURPOSE

MOVLEV REPLACEMENT WHICH CALLS MOVECM

FUNCTIONAL CATEGORIES: K2

LANGUAGE: FORTRAN IV EXTENDED

COMPUTERS (OPERATING SYSTEMS)

CDC 6000/CYBER 170 (NOS/BE)

REMARKS

AT NOS/BE LEVEL 461, THE FTN SUBROUTINE 'MOVLEV' USES CMM, WHICH CAN CAUSE PROBLEMS WITH PROGRAMS MOVING INTO PROGRAM-EXTENDED FL. SUBROUTINE 'MOVECM' IS A MUCH FASTER ROUTINE WHICH DOES NOT USE CMM, HOWEVER, IT HAS A DIFFERENT CALLING SEQUENCE. 'MOVEIT' IS A TRANSITIONAL SUBROUTINE. IT HAS THE SAME CALLING SEQUENCE AS 'MOVLEV' BUT CALLS 'MOVECM'. IT TAKES A LITTLE LONGER TO EXECUTE THE MOVE BECAUSE IT INVOLVES TWO (2) CALLS, BUT THE CALLING SEQUENCE MAY BE MORE MEANINGFUL AND EASIER TO USE.

USAGE

CALL MOVEIT (FROM, TO, NWORDS)

DESCRIPTION OF PARAMETERS

FROM - ARRAY TO BE MOVED
TO - RECEIVING ARRAY
NWORDS - NUMBER OF WORDS TO BE MOVED

CM REQUIRED: 20B

SUBPROGRAMS REQUIRED

PART OF LANGUAGE

NONE

OTHERS

MOVECM - MOVE AN ARRAY 4 WORDS AT A TIME

AUTHOR

DAVID V SOMMER - DTNSRDC CODE 1892.2

DATE WRITTEN: 10/16/79

DATE(S) REVISED

07/15/80 - MOVE TO NSRDC

02/15/83 - RECOMPILE AT NOS/BE LEVEL 552

LOCATION OF DECKS

SOURCE

UPDATE LIBRARY ON MSS: NSRDCPL,UN=CSYS

OBJECT

EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC,,MOVEIT,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'MOVSTR'

PURPOSE

MOVE A STRING OF CHARACTERS FROM ONE ARRAY TO ANOTHER

COMPUTERS (OPERATING SYSTEMS)

CDC 6000/CYBER 170 (NOS/BE)

FUNCTIONAL CATEGORIES: M4

LANGUAGE: FORTRAN IV EXTENDED

REMARKS

NONE

USAGE

CALL MOVSTR (FROM, IFROM, TO, ITO, LEN, IRC)

CALL MOVSTR (FROM, IFROM, TO, ITO, LEN)

DESCRIPTION OF PARAMETERS

FROM - ARRAY FROM WHICH STRING IS TO BE EXTRACTED
IFROM - STARTING POSITION OF STRING TO BE EXTRACTED
(POSITION 1 IS LEFT-MOST CHARACTER OF FROM(1))
TO - ARRAY TO RECEIVE THE STRING
ITO - STARTING POSITION TO INSERT THE STRING
(POSITION 1 IS LEFT-MOST CHARACTER ON TO(1))
LEN - NUMBER OF CHARACTERS IN STRING TO BE MOVED
IRC - OPTIONAL ERROR RETURN CODE
0 - NO ERROR, STRING MOVED
1 - IFROM LE 0
2 - ITO LE 0
3 - LEN LE 0

CM REQUIRED: 61B

EXAMPLE

FROM: ABCDEFGHIJKLMNOPQRSTUVWXYZ TO: *****
AFTER CALL MOVSTR (FROM, 5, TO, 12, 4, IRC)
FROM: ABCDEFGHIJKLMNOPQRSTUVWXYZ TO: *****EFGH*****
IRC : 0

SUBPROGRAMS REQUIRED

PART OF LANGUAGE

AND LOCF MOD OR SHIFT
OTHERS
NONE

AUTHOR

DAVID V SOMMER - DTNSRDC CODE 1892.2

DATE WRITTEN: 10/04/76

DATE(S) REVISED

04/04/77 - MAKE IRC OPTIONAL

02/15/83 - RECOMPILE AT NOS/BE LEVEL 552

LOCATION OF DECKS

SOURCE

UPDATE LIBRARY ON MSS: NSRDCPL,UN=CSYS

OBJECT

EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC,,MOVSTR,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'MSET'

PURPOSE

SET A SINGLE WORD (BY ABSOLUTE ADDRESS) IN USER'S FL

FUNCTIONAL CATEGORIES: K2

LANGUAGE: CDC 6000 CP COMPASS

COMPUTERS (OPERATING SYSTEMS)

CDC 6000/CYBER 170 (NOS/BE)

REMARKS

'MSET' IS AN ENTRY POINT IN 'CMDRCT'.

USAGE

CALL MSET (ADDR, NEW)

DESCRIPTION OF PARAMETERS

ADDR - ADDRESS IN USER'S FL TO BE SET

NEW - WORD TO BE PUT INTO 'ADDR'

NO ERROR CHECKING IS DONE.

CM REQUIRED: 11B (INCLUDES 'MFETCH')

SUBPROGRAMS REQUIRED

PART OF LANGUAGE

NONE

OTHERS

NONE

AUTHOR

? - NWL:

DATE WRITTEN:

DATE(S) REVISED

03/23/83 - RECOMPILE AT NOS/BE LEVEL 552

LOCATION OF DECKS

SOURCE

UPDATE LIBRARY ON MSS: NSRDCPL,UN=CSYS (*DECK CMDRCT)

OBJECT

EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC,,MSET,OUTPUT,MSACCES=<PASSWORD>.

FUNCTION 'MXGET'

PURPOSE

EXTRACT (RIGHT-JUSTIFIED, ZERO-FILLED) 0-10 6-BIT
CHARACTERS FROM 60-BIT WORDS

FUNCTIONAL CATEGORIES: M4

LANGUAGE: FORTRAN IV EXTENDED

COMPUTERS (OPERATING SYSTEMS)
CDC 6000/CYBER 170 (NOS/BE)

REMARKS

NONE

USAGE

MXGET (WORD, START, NCHAR)

DESCRIPTION OF PARAMETERS

WORD - WORD FROM WHICH CHARACTERS ARE TO BE EXTRACTED
START - STARTING CHARACTER
(LEFT-MOST CHARACTER IS POSITION 1)
NCHAR - NUMBER OF CHARACTERS TO EXTRACT (0-10)
MXGET - WILL CONTAIN ONE OF:
-1 -- START OR NCHAR OR START+NCHAR INVALID
0 -- IF NCHAR IS 0
XXX -- EXTRACTED CHARACTER STRING, R-FORMAT

CM REQUIRED: 26B

EXAMPLES

1) EXTRACT CHARACTERS 3-7 FROM A WORD CONTAINING
'ABCDEFGHIJ':
DATA WORD/ "ABCDEFGHIJ"/
...
ICHARS = MXGET (WORD, 3, 5)
ICHARS WILL CONTAIN 'CDEFG' (0000 0000 0003 0405 0607B)

2) EXTRACT 'THIS' FROM 'THISSTRING':
DATA IWORD/ "THISSTRING"/
...
IF (MXGET(IWORD,1,4) .EQ. 4RTHIS) ...

SUBPROGRAMS REQUIRED

PART OF LANGUAGE
AND SHIFT
OTHERS
NONE

AUTHOR

DAVID V SOMMER - DTNSRDC CODE 1892.2

DATE WRITTEN: 10/17/79

DATE(S) REVISED

03/23/83 - RECOMPILE AT NOS/BE LEVEL 552

LOCATION OF DECKS

SOURCE

UPDATE LIBRARY ON MSS: NSRDCPL,UN=CSYS

OBJECT

EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC,,MXGET,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'NEWDAT'

PURPOSE

ADD/SUBTRACT SPECIFIED NUMBER OF DAYS TO/FROM A GIVEN DATE

FUNCTIONAL CATEGORIES: M2

LANGUAGE: FORTRAN IV

COMPUTERS (OPERATING SYSTEMS)

CDC 6000/CYBER 170 (NOS/BE)

REMARKS

NONE

USAGE

CALL NEWDAT (FMT, OLD, NEW, OCENT, NCENT, ADD)

DESCRIPTION OF PARAMETERS

FMT - FORMAT OF DATE (INTEGER)

1 - 'MM/DD/YY '

2 - ' MM/DD/YY '

OLD - OLD DATE (MM/DD/YY)

NEW - NEW DATE

OCENT - OLD CENTURY (E.G., INTEGER 1900)

NCENT - NEW CENTURY (E.G., INTEGER 1900)

ADD - NUMBER OF DAYS TO ADD

(NEGATIVE TO SUBTRACT)

CM REQUIRED: 162B

SUBPROGRAMS REQUIRED

PART OF LANGUAGE

NONE

OTHERS

JGDATE - JULIAN/GREGORIAN DATE CONVERTER (MULTI-YEAR)

AUTHOR

DAVID V SOMMER - DTNSRDC CODE 1892.2

DATE WRITTEN: 1968

DATE(S) REVISED

02/73 - CONVERT TO SCOPE 3.3

03/21/83 - RECOMPILE AT NOS/BE LEVEL 552

LOCATION OF DECKS

SOURCE

UPDATE LIBRARY ON MSS: NSRDCPL,UN=CSYS

OBJECT

EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC,,NEWDAT,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'NFILL'

PURPOSE

FILL ELEMENTS 1 THRU N OF AN ARRAY WITH THE VALUES 1 THRU N,
RESPECTIVELY

FUNCTIONAL CATEGORIES: A1

LANGUAGE: CDC 6000 CP COMPASS

COMPUTERS (OPERATING SYSTEMS)
CDC 6000/CYBER 170 (NOS/BE)

REMARKS
NONE

USAGE

CALL NFILL (A, N)

DESCRIPTION OF PARAMETERS

A - ARRAY TO BE FILLED
N - NUMBER OF ELEMENTS TO BE FILLED

CM REQUIRED: 6B

SUBPROGRAMS REQUIRED

PART OF LANGUAGE
NONE
OTHERS
NONE

AUTHOR

DAVID V SOMMER - DTNSRDC CODE 1892.2

DATE WRITTEN: 08/09/76

DATE(S) REVISED

03/23/83 - RECOMPILE AT NOS/BE LEVEL 552

LOCATION OF DECKS

SOURCE
UPDATE LIBRARY ON MSS: NSRDCPL,UN=CSYS
OBJECT
EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC,,NFILL,OUTPUT,MSACCES=<PASSWORD>.

FUNCTION 'NFILLT'
SUBROUTINE 'NFILLT'

PURPOSE

TEST AN ARRAY FOR THE PRESENCE OF THE INTEGERS 1 THRU N
IN ELEMENTS 1 THRU N, RESPECTIVELY

FUNCTIONAL CATEGORIES: M5

LANGUAGE: FORTRAN IV EXTENDED

COMPUTERS (OPERATING SYSTEMS)
CDC 6000/CYBER 170 (NOS/BE)

REMARKS

A SUGGESTED USE OF THIS ROUTINE IS IN CONJUNCTION WITH ONE
OF THE SORTING ROUTINES TO DETERMINE IF THE ARRAY BEING
SORTED WAS ALREADY IN ORDER.

USAGE

ISUB = NFILLT (A, N, I)
CALL NFILLT (A, N, I)

DESCRIPTION OF PARAMETERS

A - ARRAY TO BE SCANNED
N - NUMBER OF ELEMENTS TO TEST
I - =0 - A(1) THRU A(N) CONTAIN 1 THRU N
 >0 - A(I) IS FIRST ELEMENT TO FAIL TEST
NFILLT - IF USED AS A FUNCTION, WILL RETURN THE SAME VALUE
 AS 'I'

CM REQUIRED: 25B

SUBPROGRAMS REQUIRED

PART OF LANGUAGE
NONE
OTHERS
NONE

AUTHOR

DAVID V SOMMER - DTNSRDC CODE 1892.2

DATE WRITTEN: 08/19/76

DATE(S) REVISED

03/23/83 - RECOMPILE AT NOS/BE LEVEL 552

LOCATION OF DECKS

SOURCE
UPDATE LIBRARY ON MSS: NSRDCPL,UN=CSYS
OBJECT
EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC,,NFILLT,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'NUMEXEC'

PURPOSE

GET NUMBER OF EXECUTE CARD PARAMETERS WHICH WERE USED IN
THIS EXECUTION OF THE PROGRAM

FUNCTIONAL CATEGORIES: Q0

LANGUAGE: FORTRAN IV

COMPUTERS (OPERATING SYSTEMS)
CDC 6000/CYBER 170 (NOS/BE)

REMARKS
NONE

USAGE
CALL NUMEXEC (NEXEC)

DESCRIPTION OF PARAMETER
NEXEC - WILL RETURN THE NUMBER OF EXECUTE CARD PARAMETERS

CM REQUIRED: 16B

METHOD
THE NUMBER OF PARAMETERS IS IN THE RIGHTMOST 18 BITS OF
WORD RA+52 (64B) IN THE USER'S FL.

SUBPROGRAMS REQUIRED
PART OF LANGUAGE
NONE
OTHERS
MFETCH - GET SPECIFIED WORD OF USER'S FL

ARITHMETIC STATEMENT FUNCTIONS
R38FMT - FAST R-FORMAT DECODE (RIGHT-ADJ, ZERO-FILLED)

AUTHOR
DAVID V SOMMER - DTNSRDC CODE 1892.2

DATE WRITTEN: 04/15/75

DATE(S) REVISED
03/23/83 - RECOMPILE AT NOS/BE LEVEL 552

LOCATION OF DECKS
SOURCE
UPDATE LIBRARY ON MSS: NSRDCPL,UN=CSYS
OBJECT
EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT
BEGIN,DOCGET,,NSRDC,,NUMEXEC,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'NUMVAR'

PURPOSE

GET THE NUMBER OF ARGUMENTS THAT WERE PASSED TO THE ROUTINE
WHICH CALLED NUMVAR

FUNCTIONAL CATEGORIES: Q0

LANGUAGE: CDC 6000 CP COMPASS

COMPUTERS (OPERATING SYSTEMS)
CDC 6000/CYBER 170 (NOS/BE)

REMARKS

WHEN USED, IT SHOULD PRECEDE OTHER EXECUTABLE STATEMENTS IN
THE SUBPROGRAM TO INSURE THAT THE REGISTERS HAVE NOT BEEN
DESTROYED.

USAGE

CALL NUMVAR (NARGS)

DESCRIPTION OF PARAMETER

NARGS - WILL CONTAIN THE NUMBER OF ARGS IN THE ROUTINE
WHICH CALLED NUMVAR

CM REQUIRED: 5B

SUBPROGRAMS REQUIRED

PART OF LANGUAGE

NONE

OTHERS

NONE

AUTHOR

MIKE CHERNICK

DATE WRITTEN: UNKNOWN

DATE(S) REVISED

03/22/83 - RECOMPILE AT NOS/BE LEVEL 552

LOCATION OF DECKS

SOURCE

UPDATE LIBRARY ON MSS: NSRDCPL,UN=CSYS

OBJECT

EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC,,NUMVAR,OUTPUT,MSACCES=<PASSWORD>.

FUNCTION 'OFMTDE'

PURPOSE

FAST O-FORMAT DECODE

FUNCTIONAL CATEGORIES: 12

LANGUAGE: FORTRAN IV

COMPUTERS (OPERATING SYSTEMS)

CDC 6000/CYBER 170 (NOS/BE)

REMARKS

NONE

USAGE

VARIABLE = OFMTDE (IWORD, ISTART, NCHAR)

DESCRIPTION OF PARAMETERS

VARIABLE - WILL CONTAIN THE RESULT RIGHT-JUSTIFIED
OR -1 IF NON-OCTAL DIGIT FOUND
OR -2 IF ISTART IS OUT OF RANGE
OR -3 IF ISTART+NCHAR GREATER THAN 10.
(IF VARIABLE IS INTEGER, OFMTDE MUST BE DECLARED
INTEGER IN THE CALLING PROGRAM)

IWORD - WORD FROM WHICH THE FIELD WILL BE EXTRACTED

ISTART - FIRST CHARACTER POSITION OF FIELD WITHIN IWORD
(1-10)

NCHAR - NUMBER OF CHARATERS IN FIELD (1-10)
(ISTART+NCHAR MUST BE LESS THAN 11)

CM REQUIRED: 51B

EXAMPLE

VARIABLE = OFMTDE (10L1234567654, 6, 3) WILL PRODUCE
VARIABLE = 0000 0000 0000 0000 0676B

VARIABLE = OFMTDE (5L123.4, 3, 3) WILL PRODUCE
VARIABLE = 7777 7777 7777 7777 7776B

VARIABLE = OFMTDE (IWORD, 0, 5) WILL PRODUCE
VARIABLE = 7777 7777 7777 7777 7775B

VARIABLE = OFMTDE (IWORD, 3, 8) WILL PRODUCE
VARIABLE = 7777 7777 7777 7777 7774B

SUBPROGRAMS REQUIRED
PART OF LANGUAGE
SHIFT
OTHERS
NONE

AUTHOR
DAVID V SOMMER - DTNSRDC CODE 1892.2

DATE WRITTEN: 11/24/75

DATE(S) REVISED
03/23/83 - RECOMPILE AT NOS/BE LEVEL 552

LOCATION OF DECKS
SOURCE
UPDATE LIBRARY ON MSS: NSRDCPL,UN=CSYS
OBJECT
EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT
BEGIN,DOCGET,,NSRDC,,OFMTDE,OUTPUT,MSACCES=<PASSWORD>.

FUNCTION 'OFMTV'

PURPOSE

FAST O-FORMAT DECODE OF VARIABLE LENGTH INPUT

FUNCTIONAL CATEGORIES: 12

LANGUAGE: FORTRAN IV

COMPUTERS (OPERATING SYSTEMS)
CDC 6000/CYBER 170 (NOS/BE)

REMARKS
NONE

USAGE

VARIABLE = OFMTV (I)

DESCRIPTION OF PARAMETERS

VARIABLE - WILL CONTAIN THE RESULT RIGHT-JUSTIFIED
OR -1 IF A NON-OCTAL DIGIT FOUND.
IF VARIABLE IS INTEGER, OFMTV MUST BE
DECLARED INTEGER IN THE CALLING PROGRAM.
I - WORD OF OCTAL DIGITS ENDING WITH AN OCTAL
00B. (EG, 3L123, 9L123456701)

CM REQUIRED: 27B

EXAMPLE

VARIABLE = OFMTV (5L12345) WILL RETURN
VARIABLE = 0000 0000 0000 0001 2345B

VARIABLE = OFMTV (1L+) WILL RETURN
VARIABLE = 7777 7777 7777 7777 7776B

SUBPROGRAMS REQUIRED

PART OF LANGUAGE
SHIFT
OTHERS
NONE

AUTHOR

DAVID V SOMMER - DTNSRDC CODE 1892.2

DATE WRITTEN: 11/24/75

DATE(S) REVISED

03/23/83 - RECOMPILE AT NOS/BE LEVEL 552

LOCATION OF DECKS

SOURCE
UPDATE LIBRARY ON MSS: NSRDCPL,UN=CSYS
OBJECT
EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC,,OFMTV,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'OMRONI'

PURPOSE

INITIALIZE COMMON BLOCK /OMRON/ WITH ASCII CONTROL CODES
FOR OMRON CRT'S

FUNCTIONAL CATEGORIES: M4

LANGUAGE: FORTRAN 77 EXTENDED

COMPUTERS (OPERATING SYSTEMS)
CDC 6000/CYBER 170 (NOS/BE)

REMARKS

THIS SUBROUTINE MUST BE EXECUTED PRIOR TO GENERATING ASCII
MESSAGES WITH OMRON CONTROL CODES USING SUBROUTINE ASCII.

COMMON BLOCK /OMRON/ IS OBTAINED BY RUNNING PROCEDURE ASCII0
AND INSERTING THE COMMON BLOCK INTO EACH (SUB)PROGRAM WHICH
WILL GENERATE ASCII MESSAGES HAVING OMRON CONTROL CODES.

USAGE

CALL OMRONI

CM REQUIRED: 6B

NAMES OF CONTROL CODES

TAB CONTROL

TABSET - SET TAB
TABCLR - CLEAR TAB
TAB - TAB
TABACK - BACK TAB

CURSOR CONTROL

CURSU - CURSOR UP
CURSD - CURSOR DOWN
CURSR - CURSOR RIGHT
CURSL - CURSOR LEFT
HOME - CURSOR HOME

SCREEN CONTROL

SCROLD - SCROLL DOWN
SCROLU - SCROLL UP
NXTPAG - NEXT PAGE
PRVPAG - PREVIOUS PAGE

INSERT/DELETE

INSCH - INSERT CHARACTER
INSLIN - INSERT LINE
DELCH - DELETE CHARACTER
DELIN - DELETE LINE

CLEAR

CLRLIN - CLEAR LINE
CLRSCR - CLEAR SCREEN
CLRMEM - CLEAR MEMORY

VIDEO

VIDNRM - NORMAL VIDEO
VIDOFF - VIDEO OFF
VIDREV - REVERSE VIDEO
VIDDIM - DIM VIDEO
REVDIM - DIM REVERSE VIDEO
BLINK - BLINK
BLNKRV - BLINK REVERSE
UNDRON - UNDERLINE ON

SUBPROGRAMS REQUIRED

PART OF LANGUAGE

NONE

OTHERS

NONE

AUTHOR

DAVID V SOMMER - DTNSRDC CODE 1892.2

DATE WRITTEN: 03/07/84

DATE(S) REVISED

LOCATION OF DECKS

SOURCE DECK

UPDATE LIBRARY ON MSS: NSRDCPL,UN=CSYS

OBJECT DECK

EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC,,OMRONI,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'OPLSA'

PURPOSE

ORTHOGONAL POLYNOMIAL LEAST SQUARE APPROXIMATION

FUNCTIONAL CATEGORIES: E2

LANGUAGE: FORTRAN IV EXTENDED

COMPUTERS (OPERATING SYSTEMS)

CDC 6000/CYBER 170 (NOS/BE)

REMARKS

THE APPROXIMATING POLYNOMIAL IS

$$C(1)+C(2)*X+C(3)*X**2+...+C(M+1)*X**M$$

FOR MORE THAN 9TH DEGREE OR MORE THAN 30 DATA POINTS, THE
SOURCE PROGRAM MUST BE REDIMENSIONED.

USAGE

CALL OPLSA (N, W, X, F, M, D, A, C)

DESCRIPTION OF PARAMETERS

N - NUMBER OF DATA POINTS (MAX: 30)

W - ARRAY OF N WEIGHTS

X - ARRAY OF N DATA POINTS

F - ARRAY OF N FUNCTION VALUES

M - DESIRED DEGREE OF POLYNOMIAL (MAX: 9)

D - OUTPUT ARRAY OF COEFFICIENTS OF POLYNOMIALS $O(J,X)$
(DIMENSION: 10,N)

A - OUTPUT ARRAY OF COEFFICIENTS OF $O(J,X)$ 'S OF LEAST
SQUARE POLYNOMIALS (DIMENSION: M+1)

C - ARRAY TO CONTAIN COEFFICIENTS OF RESULTING LEAST SQUARE
POLYNOMIAL (SEE REMARKS) (DIMENSION: M+1)

CM REQUIRED: 767B

SUBPROGRAMS REQUIRED
PART OF LANGUAGE
NONE
OTHERS
NONE

AUTHORS
UNIVERSITY OF MARYLAND
S VOIGT

DATE WRITTEN: 1971

DATE(S) REVISED
09/20/83 - RECOMPILE AT NOS/BE LEVEL 552

LOCATION OF DECKS
SOURCE
UPDATE LIBRARY ON MSS: NSRDCPM,UN=CSYS (*DECK AMOPLSA)
OBJECT
EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT
BEGIN,DOCGET,,NSRDC,,OPLSA,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'OVLNAME'

PURPOSE

GET NAME OF FILE CURRENTLY BEING EXECUTED

FUNCTIONAL CATEGORIES: Q0

LANGUAGE: CDC 6000 CP COMPASS

COMPUTERS (OPERATING SYSTEMS)
CDC 6000/CYBER 170 (NOS/BE)

REMARKS

'I' MAY BE USED AS THE FIRST ARGUMENT IN 'CALL OVERLAY'.

USAGE

CALL OVLNAME (I)

DESCRIPTION OF PARAMETER

I - WILL CONTAIN THE LOCAL FILE NAME CURRENTLY BEING
EXECUTED

CM REQUIRED: 3

METHOD

THE FILE NAME IS EXTRACTED FROM BITS 59-18 OF WORD
RA+64B IN THE USER'S FIELD LENGTH

SUBPROGRAMS REQUIRED

PART OF LANGUAGE
NONE
OTHERS
NONE

AUTHOR

? - NWL

DATE WRITTEN: ?

DATE(S) REVISED

03/22/83 - RECOMPILE AT NOS/BE LEVEL 552

LOCATION OF DECKS

SOURCE

UPDATE LIBRARY ON MSS: NSRDCPL,UN=CSYS

OBJECT

EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC,,OVLNAME,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'PARGET'

PURPOSE

GET ALL PARAMETERS OF USER-SUPPLIED PARAMETER STRING

FUNCTIONAL CATEGORIES: M4

LANGUAGE: FORTRAN IV EXTENDED

COMPUTERS (OPERATING SYSTEMS)

CDC 6000/CYBER 170 (NOS/BE)

REMARKS

NONE

USAGE

CALL PARGET (IAREA, LAREA, IPARAM, NPARAM, ISEP, RSEP, LSEP)

CALL PARGET (IAREA, LAREA, IPARAM, NPARAM, ISEP, RSEP)

CALL PARGET (IAREA, LAREA, IPARAM, NPARAM, ISEP)

CALL PARGET (IAREA, LAREA, IPARAM, NPARAM)

DESCRIPTION OF PARAMETERS

IAREA - AREA CONTAINING PARAMETER LIST TO BE EXTRACTED

LAREA - NUMBER OF WORDS IN 'IAREA' (16 MAX)

IPARAM - ARRAY TO CONTAIN PARAMETERS

(IF IT IS NOT KNOWN WHETHER THE PARAMETER LIST IN IAREA CONTAINS A TERMINATOR ('.' OR ')') OR NOT, THEN IPARAM, ISEP, LSEP AND RSEP SHOULD BE DIMENSIONED AT LEAST 10 TIMES LAREA. THIS WILL ALLOW FOR THE WORST POSSIBLE CASE (IAREA ALL BLANKS).)

NPARAM - WILL BE NUMBER OF PARAMETERS FOUND

ISEP - IF PRESENT, ARRAY TO CONTAIN A CODE IDENTIFYING THE SEPARATOR FOUND FOLLOWING THE CORRESPONDING PARAMETER

DEC	OCT	SEPARATOR
-----	-----	-----------

1	1	,
---	---	---

2	2	=
---	---	---

3	3	/
---	---	---

4	4	(
---	---	---

5	5	+
---	---	---

6	6	-
---	---	---

7	7	BLANK
---	---	-------

8	10B	;
---	-----	---

14	16B	OTHER
----	-----	-------

15	17B	. OR) (TERMINATOR)
----	-----	---------------------

RSEP - IF PRESENT, ARRAY TO CONTAIN THE SEPARATOR FOUND (1R FORMAT)

LSEP - IF PRESENT, ARRAY TO CONTAIN THE SEPARATOR FOUND (1L FORMAT)

CM REQUIRED: 117B

SUBPROGRAMS REQUIRED
PART OF LANGUAGE
LOCF
OTHERS
EXTPRM - EXTRACT THE NEXT PARAMETER

AUTHOR
DAVID V SOMMER - DTNSRDC CODE 1892.2

DATE WRITTEN: 04/11/74

DATE(S) REVISED
11/18/75 - NAME CHANGED FROM GETPAR TO PARGET TO AVOID
CONFLICT WITH SYSIO ROUTINE OF SAME NAME
06/24/76 - PROCESSING OF OPTIONAL PARAMETERS MODIFIED
03/21/83 - RECOMPILE AT NOS/BE LEVEL 552

LOCATION OF DECKS
SOURCE
UPDATE LIBRARY ON MSS: NSRDCPL,UN=CSYS
OBJECT
EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT
BEGIN,DOCGET,,NSRDC,,PARGET,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'PF'
SUBROUTINE 'PFX'
SUBROUTINE 'PFLook'
SUBROUTINE 'PFWAIT'
SUBROUTINE 'LF'

PURPOSE

FORTRAN CALLABLE PERMANENT FILE FUNCTIONS AND AUXILIARY FILE
ACTION REQUESTS

FUNCTIONAL CATEGORIES: Q3

LANGUAGE: CDC 6000 CP COMPASS

COMPUTERS (OPERATING SYSTEMS)
CDC 6000/CYBER 170 (NOS/BE)

REMARKS

THE PERMANENT FILE FUNCTIONS HAVE BEEN DEVELOPED ACCORDING
TO A CDC WRITTEN AND VIM APPROVED SPECIFICATION. THE
AUXILIARY FILE ACTION REQUESTS HAVE BEEN EXTENDED TO INCLUDE
THE 'ROUTE' FUNCTION.

SINCE IT IS NOT KNOWN IF CDC WILL RELEASE THE CODE, THIS
VERSION WAS OBTAINED FROM VIM.

EXTENSIONS TO THE APPROVED SPECIFICATION ARE NOTED IN THE
DESCRIPTION BELOW.

ONLY THOSE PARAMETERS WHICH HAVE MEANING AT DTNSRDC ARE
LISTED IN THIS DOCUMENT.

USERS SHOULD EXPERIENCE LITTLE OR NO PROGRAM MODIFICATION
IF CDC IMPLEMENTS THIS FACILITY.

NOTES FOR COBOL:

1. THE PARAMETERS FOR THE PF FUNCTIONS SHOULD BE AT THE
BEGINNING OF WORKING STORAGE TO ENSURE CORRECT WORD
BOUNDARY.
2. PICTURES MUST BE FULL WORDS.
3. THE PFN MUST BE A PICTURE OF FOUR WORDS----X(40).
4. RCV AND ERV MUST BE COMP-1 PIC 9(10).
5. REMEMBER TO ENTER COMPASS, NOT FORTRAN-X.
6. WHEN ENTERING PF, REMEMBER TO PUT PF IN QUOTES SINCE
IT IS A COBOL RESERVED WORD.
7. BOTH LFN AND PFN MUST BE SPECIFIED.

THIS PACKAGE REPLACES THE FOLLOWING SUBROUTINES IN LIBRARY
NSRDC: REQUEST, ROUTE, ZPFPUT, ZPFUNC, ZRTPUT. THESE
ROUTINES WILL CONTINUE TO BE AVAILABLE FOR USE BY EXISTING
PROGRAMS, BUT MAY NOT BE UPGRADED FOR FUTURE RELEASES OF
NOS/BE.

RELATED ROUTINES IN LIBRARIES NSRDC AND/OR NSRDC5:

LPPFERR - SEPARATE LF/PF "ERR" CODES
PFR - INTERPRET PF INTEGER RETURN CODE ("RC")
ROUTERC - INTERPRET LF("ROUTE") INTEGER RETURN CODE ("RC")

USAGE

```
CALL PF ("FUNCTION", "LFN", "PFN", KEY(1), VAL(1), ...,
        KEY(N), VAL(N))
CALL PFX ("FUNCTION", "LFN", "PFN", KEYS, VALS, RETURN)

CALL PFLOOK (LOOK)
CALL PFWAIT (LOOK)

CALL LF ("REQUEST", "LFN", "KEY(1)", "KEY(2)", "VAL(1)",
        ... )

CALL LF ("RETURN", "LFN(1)", "LFN(2)", ..., "LFN(N)")
CALL LF ("RETURNX", "LFN(1)", "LFN(2)", ..., "LFN(N)")

CALL LF ("ROUTE", "LFN", "KEY(1)", "KEY(2)", "VAL(1)", ... )
```

DESCRIPTION OF PARAMETERS

CALL PF

FUNCTION - ONE OF THE FOLLOWING:

FTN5/4	FTN5	FTN4
"ATTACH"	OR L"ATTACH"	OR 6LATTACH
"CATALOG"	OR L"CATALOG"	OR 7LCATALOG
"EXTEND"	OR L"EXTEND"	OR 6LEXTEND
"RENAME"	OR L"RENAME"	OR 6LRENAME
"PURGE"	OR L"PURGE"	OR 5LPURGE
"ALTER"	OR L"ALTER"	OR 5LALTER
"PERM"	OR L"PERM"	OR 4LPERM
(PERM IS EXTENSION)		

LFN - LOCAL FILE NAME
(LEFT-JUSTIFIED DISPLAY CODE (FIRST MUST BE ALPHABETIC, FIRST 7 CHARACTERS USED)
--OR--
N (FOR TAPEN, 1 <= N <= 999))

PFN - PERMANENT FILE NAME
(0-40 LEFT-JUSTIFIED DISPLAY CODE,
FIRST BLANK OR BINARY ZERO ENDS THE PFN)
(EXTENSION: IF FIRST CHARACTER OF PFN IS BLANK OR BINARY ZERO, PFN = LFN)
(EXTENSION: IF FIRST CHARACTER OF LFN IS BLANK OR BINARY ZERO, LFN = PFN (FIRST 7 CHARACTERS))

KEYI, VALI - ONE OF THE FOLLOWING PAIRS:

	KEY(I)	VAL(I)
	"ID"	4- TO 9-CHARACTER STRING
	"RP"	3-DIGIT STRING
*RW	"CY"	3-DIGIT STRING
	"TK"	9-CHARACTER STRING
	"CN"	9-CHARACTER STRING
	"MD"	9-CHARACTER STRING
	"EX"	9-CHARACTER STRING
	"RD"	9-CHARACTER STRING
	"XR"	9-CHARACTER STRING

"PW" UP TO 5 PASSWORDS
 9-CHARACTER STRING
 "AC" 10-CHARACTER STRING
 (ALL DIGITS OR S+9 DIGITS)
 "MR" "0" OR "1"
 "LC" "0" OR "1"
 "RW" "0" OR "1"
 "RB" "0" OR "1"
 "FO" "IS" OR "DA" OR "AK"
 "ST" 3-CHARACTER STRING
 "SN" 7-CHARACTER STRING
 (FOR USER DEVICE SET)
 (SYNONYM: "SETNAME")
 "VSN" 6-CHARACTER STRING
 (FOR USER DEVICE SET)
 *W "RC" INTEGER VARIABLE FOR RETURN CODE
 (EXTENSION -- NOT FOR COBOL4)
 *W "RRC" REAL VARIABLE FOR RETURN CODE
 (EXTENSION)
 "NONE" ANY VALUE (SKIP NEXT WORD IN
 PARAMETER LIST - NO OP)
 *W "ERR" INTEGER VARIABLE CONTAINING ERROR
 CODE FOR PARAMETER ERRORS. USE AS
 FIRST KEY(I) WHEN DEBUGGING, THEN
 REMOVE AFTER DEBUGGING IS
 COMPLETE.
 (SEE 'FORMAT OF ERR' BELOW.)
 "NR" "0" OR "1" (EXTENSION)
 "0" - DO NOT RETURN UNTIL ACTION
 HAS COMPLETED.
 "1" - START ACTION AND DON'T WAIT.
 YOU ARE RESPONSIBLE FOR
 CHECKING THE COMPLETION BY
 USING PFLOOK AND/OR PFWAIT.
 "DAY" DAYFILE OUTPUT (EXTENSION)
 "ON" - PUT 1- TO 5-LINE SUMMARY
 INTO DAYFILE
 "OFF" - NO DAYFILE MESSAGES
 "FLUSH" - DISPLAY CURRENT DAYFILE
 MESSAGES NOW
 (DEFAULT: "ON")
 (CAUTION: IF DAY=OFF, ERROR
 MESSAGES ARE ALSO SUPPRESSED)

 *RW - VAL(I) IS READ/WRITE. MUST BE A
 VARIABLE, NOT A CONSTANT.
 *W - VAL(I) IS WRITE ONLY. MUST BE A
 VARIABLE, NOT A CONSTANT.
 ALL OTHERS ARE READ ONLY.

CALL PFX (LIKE PF, EXCEPT KEYS AND VALUES ARE IN ARRAYS)

FUNCTION - SAME AS FOR 'PF'
 LFN - SAME AS FOR 'PF'
 PFN - SAME AS FOR 'PF'
 KEYS - ARRAY CONTAINING KEYWORDS.
 (END WITH "END")
 VALS - ARRAY CONTAINING CORRESPONDING VALUES.
 RETURN - RETURN CODE
 (WILL HAVE SAME AS "RC" PARAMETER)

CALL PFLOOK (CHECK ON PERMANENT FILE STATUS)

LOOK - WILL CONTAIN THE STATUS IN THE FORMAT:

BITS	EXPLANATION
59	0 = PREVIOUS FUNCTION INCOMPLETE 1 = PREVIOUS FUNCTION COMPLETE
58:9	0 - UNUSED
8:0	ERROR VALUE (SEE 'NOS/BE-GENERATED RETURN CODES FOR PF/PFX' BELOW)

CALL PFWAIT (WAIT UNTIL PERMANENT FILE FUNCTION IS COMPLETE)

LOOK - SAME AS FOR 'PFLOOK'

NOTE - IF THE ACTION WILL NEVER COMPLETE, THE SYSTEM WILL
ABORT THE JOB WITH 'JOB HUMG IN AUTO RECALL'. THE
CODE IN THE VICINITY OF THIS SUBROUTINE HAS
PROBABLY BEEN CLOBBED.

CALL LF ("REQUEST", ...) (REQUEST *PF OR *Q)

LFN - SAME AS FOR 'PF'

KEY(I) - ONE OF:

- "*PF" - PERMANENT FILE SPACE
- "*Q" - QUEUE SPACE
- "SN" - SETNAME OF USER DEVICE SET
(THE NEXT PARAMETER IS THE <SETNAME>)
- "VSN" - VOLUME SERIAL NUMBER OF USER DEVICE
SET
(THE NEXT PARAMETER IS THE <VSN>)

VAL(I) - VALUES FOR "SN" AND "VSN" KEYS

CALL LF ("RETURN", ...) (RETURN FILES)

CALL LF ("RETURNX", ...)

LFN(I) - SAME AS FOR 'PF'

NOTES - 1) RETURN RETURNS ANY FILE IN THE PROGRAM,
WHETHER IT HAS BEEN DECLARED OR NOT
(LOOKS FOR FIT'S (FILE INFORMATION TABLES))
RETURNX RETURNS ANY FILE, EVEN IF IT IS NOT
PART OF THE PROGRAM (DOES NOT LOOK FOR FIT'S).

- 2) THIS SETS THE FIT (IF THERE IS ONE) AS BEING
CLOSED. THIS IS TO PREVENT RECORD MANAGER
FROM RE-OPENING THE FILE AT END-OF-PROGRAM.
BE CAREFUL! IF YOUR FILE HAS NOT BEEN CLOSED
PROPERLY, LF(RETURN) WILL NOT FLUSH YOUR
BUFFERS AND WILL NOT FIX UP ANY DIRECTORIES
OF OTHER FILE LINKAGES. THEREFORE, BE SURE
YOUR FILES ARE CLOSED. 'REWIND', 'END FILE',
'CLOSE', 'CALL CLOSEM', 'CALL CLOSMS' ARE
SOME WAYS TO CLOSE A FILE (USE ONE FOR THE
KIND OF FILE TO BE CLOSED) BEFORE LF(RETURN).

CALL LF ("ROUTE", ...) (ROUTE A FILE)

LFN - SAME AS FOR 'PF'

KEY(I) - ONE OF:

"ERR"

- SAME AS FOR 'PF'

"RC"

- SAME AS FOR 'PF'

(SEE 'NOS/BE-GENERATED
MESSAGES FOR LF(ROUTE)'
BELOW)

"RRC"

- SAME AS FOR 'PF'

"FID"

- FILE ID

(1-5 DISPLAY CODE CHARACTERS)
OPTIONALLY PRECEDED BY *)

"TID"

- TERMINAL ID

(3 HEXADECIMAL DIGITS, -OR-
"C" FOR CENTRAL SITE)

*S "DEFER"

- DON'T ROUTE THE FILE UNTIL
END-OF-JOB

(SYNONYMS: "DEF", "DEFERRED")

"DC"

- DISPOSITION CODE. ONE OF:

"SC" - SCRATCH (DEFAULT)

"PR" - PRINTER

"PU" - PUNCH

"IN" - INPUT QUEUE

"TO"

- COBOLESE SYNONYM FOR "DC"

(SYNONYMS: "EQUIP",
"EQUIPMENT")

NOTE: "DC" WORKS ONLY WITH
THE ABOVE KEYWORDS;
"TO", ETC., WORK ONLY
WITH THE FOLLOWING
KEYWORDS:

"PRINTER" - PRINTER

"PRINT" - PRINTER

"OUTPUT" - PRINTER

"PUNCH" - PUNCH

"PUNCHB" - PUNCH

"SCRATCH" - SCRATCH

"INPUT" - INPUT QUEUE

"FC"

- FORMS CODE (SEE CCRM, P 2-14)
(SYNONYM: "FORMS")

"REP"

- REPEAT COUNT FOR NUMBER OF
OUTPUT COPIES

(SYNONYMS: "REPETCOUNT",
"REPEATCNT")

"DEPENDENCY" - DEPENDENCY COUNT

"PRI" - PRIORITY FOR REMOTE SITES
(SYNONYM: "PRIORITY")
"EXT" - EXTERNAL CHARACTERISTICS CODE
(SEE CCRM, P 2-11)
(NOTE: "EXT" USED TO AVOID
CONFLICT WITH "EC"
PARAMETER OF PF/PFX,
WHICH IS NOT USED AT
DTNSRDC.)
(SYNONYMS: "EXTERNAL",
"EXTCODE")
"RETURNNAME" - RETURN NAME ASSIGNED BY THE
SYSTEM
"SC" - SPACING CODE FOR 580 PRINTER
ON CYBER 176
(SYNONYM: "SPACECODE")

*S - SINGLE KEYWORD. ALL OTHERS REQUIRE A
VAL(I) AS THE NEXT PARAMETER.

VAL(I) - REQUIRED FOR THOSE KEYWORDS WHICH HAVE VALUES
AND MUST FOLLOW THE CORRESPONDING KEY(I)

CM REQUIRED: 2060B

EXAMPLES

1) ATTACH, COPY AND CATALOG A PERMANENT FILE OF CARD IMAGES:

```

PROGRAM COPY80 (INPUT, OUTPUT)
C
C   SAMPLE PROGRAM TO COPY A FILE OF CARD IMAGES
C
C   CHARACTER REC * 80
C
C   INTEGER PFNIN(4), PFNOUT(4)
C   INTEGER RC
C
C   GET NAME OF FILE TO BE COPIED
C
100 CONTINUE
   IDIN = " "
   READ (*, 1, END=110) IDIN, PFNIN
C
C   IF SPECIFIED, ATTACH THE FILE
C
110 CONTINUE
   IF (IDIN .NE. " ") THEN
      CALL PF ("ATTACH", "INFYL", PFNIN, "ID", IDIN,
A          "RC", RC)
C
C   IF NOT ATTACHED, PRINT ERROR MESSAGE
C
C   IF (RC .NE. 0) THEN
      WRITE (*, 2) 'ATTACH', PFNIN, IDIN, RC
C
C   FILE ATTACHED, GET PF SPACE AND COPY IT
C
      ELSE
        CALL LF ("REQUEST", "OUTFYL", "*PF")
        OPEN (1, FILE='INFYL')
        OPEN (2, FILE='OUTFYL')
        NIN = 0
120      CONTINUE
        READ (1, 3, END=130) REC
        NIN = NIN + 1
        WRITE (2, 3) REC
        GO TO 120
C
C   WRITE AN EOF ON OUTPUT FILE
C
130      CONTINUE
        END FILE 2
        WRITE (*, 4) NIN
C
C   GET NAME OF NEW FILE (IF NONE, USE OLD PFN/ID)
C
        IDOUT = " "
        READ (*, 1, END=140) IDOUT, PFNOUT

```

```

C
C      CATALOG THE NEW FILE
C
140  CONTINUE
      IF (IDOUT .EQ. " ") THEN
          CALL PF ("CATALOG", "OUTFYL", PFNIN,
A              "ID", IDIN, "RC", RC)
          IF (RC .NE. 0) WRITE (*, 2) 'CATALOG', PFNIN,
A              IDIN, RC
          ELSE
              CALL PF ("CATALOG", "OUTFYL", PFNOUT,
A              "ID", IDOUT, "RC", RC)
              IF (RC .NE. 0) WRITE (*, 2) 'CATALOG', PFNOUT,
A              IDOUT, RC
          END IF
      END IF
C
C      ERROR - NO FILE NAME READ
C
      ELSE
          WRITE (*, 5)
      END IF
C
C      RETURN FILES
C      (AN ALTERNATE WAY IS:
C      REWIND 2
C      CALL LF ("RETURN", "INFYL", "OUTFYL"))
C
      CLOSE (1, STATUS='DELETE')
      CLOSE (2, STATUS='DELETE')
C
C      FORMATS
C
1  FORMAT (A4, 4A10)
2  FORMAT ('O COULD NOT ', A, ' FILE ', 4A10 /
A      ' ID=', A4, ' - RETURN CODE = ', O3, 'B')
3  FORMAT (A)
4  FORMAT ('O', I10, ' RECORDS COPIED')
5  FORMAT ('O ERROR - NO FILE NAME READ - COPY NOT DONE')
C
      END

```

2) CREATE AND ROUTE A FILE TO PRINT ON THE XEROX PRINTER:

```
PROGRAM ROUTST
C
C SAMPLE PROGRAM ILLUSTRATING ROUTE
C
C CHARACTER LINE * 133
C CHARACTER RCTEXT * 50
C
C INTEGER RC
C
C GET QUEUE SPACE FOR FILE
C
100 CONTINUE
CALL LF ("REQUEST", "XRX", "*Q")
OPEN (1, FILE='XRX')
REWIND 1
C
C CODE TO WRITE TO FILE XRX
C
C ...
C
C CLOSE AND ROUTE
C
REWIND 1
CLOSE (1, STATUS='KEEP')
CALL LF ("ROUTE", "XRX", "DC", "PR", "TID", "777",
A      "FC", "HH",
B      "FID", "ADS", "RETURNAME", NAME, "RC", RC)
C
C PRINT MESSAGE      (ROUTERC IS IN LIBRARY NSRDC5)
C
C IF (RC .NE. 0) THEN
C   CALL ROUTERC (RC, RCTEXT)
C   PRINT 1, RC, RC, RCTEXT
C ELSE
C   PRINT 2, NAME
C END IF
C
C FORMATS
C
1 FORMAT ('OROUTE FAILED - RETURN CODE = ', O3, 'B (',
A      I2, ') ' /
B      1X, A)
2 FORMAT ('OFILE ROUTED WITH NAME ', A7)
C
END
```


3) COPY AND CATALOG A FILE OF CARD IMAGES USING COBOL5:

IDENTIFICATION DIVISION.

PROGRAM-ID. CPYFYL.

ENVIRONMENT DIVISION.

INPUT-OUTPUT SECTION.

FILE-CONTROL.

SELECT IN-FILE ASSIGN TO "INPUT".

SELECT OUT-FILE ASSIGN TO "TEST"

USE "BT=C,RT=Z".

DATA DIVISION.

FILE SECTION.

FD IN-FILE

LABEL RECORDS ARE OMITTED

DATA RECORD IS IN-REC.

01 IN-REC PIC X(80).

FD OUT-FILE

LABEL RECORDS ARE OMITTED

DATA RECORD IS OUT-REC.

01 OUT-REC PIC X(80).

WORKING-STORAGE SECTION.

01 REQUEST PIC X(10) VALUE "REQUEST" .

01 PF-SPACE PIC X(10) VALUE "*PF" .

01 CATALOG PIC X(10) VALUE "CATALOG" .

01 LFN PIC X(10) VALUE "TEST" .

01 PFN PIC X(10) VALUE "TESTPFN" .

01 PFNFIL PIC X(30) VALUE SPACES.

01 ERR PIC X(10) VALUE "ERR" .

01 ERRV COMP-1 PIC 9(10).

01 ID PIC X(10) VALUE "ID" .

01 IDV PIC X(10) VALUE "XXXX" .

01 RC PIC X(10) VALUE "RC" .

01 RCV COMP-1 PIC 9(10).

01 OUT-LINE PIC X(80).

PROCEDURE DIVISION.

START-PG.

OPEN INPUT IN-FILE.

ENTER COMPASS "LF" USING REQUEST LFN PF-SPACE.

OPEN OUTPUT OUT-FILE.

READ-PARAG.

READ IN-FILE

AT END GO TO EOJ.

MOVE IN-REC TO OUT-LINE.

WRITE OUT-REC FROM OUT-LINE.

GO TO READ-PARAG.

EOJ.

CLOSE IN-FILE OUT-FILE.

ENTER COMPASS "PF" USING CATALOG LFN PFN ERR ERRV

ID IDV RC RCV.

IF RCV IS EQUAL TO 0

DISPLAY "FILE CATALOGUED"

ELSE

DISPLAY "FILE NOT CATALOGUED".

STOP RUN.

ERROR MESSAGES

IF "ERR" IS PROVIDED, PARAMETERS ERRORS ARE RETURNED (SEE
FORMAT OF "ERR").

IF "RC" OR "RRC" IS PROVIDED, EXECUTION ERRORS ARE
RETURNED (SEE NOS/BE-GENERATED MESSAGES).

OTHER ERROR MESSAGES INCLUDE:

ENTERED FROM 000121 - INTERNAL PF ABORT
SOME OTHER ERROR OCCURRED.

NO ERROR RETURN SUPPLIED - INTERNAL PF ABORT
SOME PARAMETER WAS IN ERROR AND "ERR" WAS NOT USED.

FORMAT OF "ERR"

BITS	EXPLANATION
59:36	0 - UNUSED
35:18	PARAMETER IN ERROR
17:0	SPECIFIC ERROR

SUBROUTINE LPPFERR IN LIBRARY NSRDC MAY BE USED TO EXTRACT
THESE TWO VALUES.

SPECIFIC ERRORS

ROUTINE	FUNCTION	OCT	DEC	EXPLANATION
		0001	1	INTERDICTED PROCESSING - UNEXPECTED BINARY ZERO PARAMETER. ALL ROUTINES USE THIS COMMON EXIT - LOOK FOR CORE CLOBBÈR OR MESSED UP PARAMETER LIST OR SHORT LIST.
LF		0010	8	NO PARAMETERS SUPPLIED AT ALL.
LF		0020	16	FUNCTION IS BINARY ZERO.
LF		0030	24	UNDEFINED FUNCTION.
LF	"ROUTE"	0050	40	NO FILE NAME.
LF	"ROUTE"	0060	48	FILE NAME BINARY ZERO.
LF	"ROUTE"	0070	56	UNDEFINED FUNCTION.
LF	"REQUEST"	0100	64	NO FILE NAME.
LF	"REQUEST"	0110	72	UNDEFINED PARAMETER.
LF	"REQUEST"	0120	80	1ST CHAR LFN NO-ALFA.
PFX		0200	128	BINARY ZERO ADDRESS IN 4TH PARAMETER. NOTE THAT IN THIS CASE THE PARAM NUMBER IS GARBAGE.
PFX		0210	136	BINARY ZERO ADDRESS IN 5TH PARAMETER. NOTE THAT IN THIS CASE THE PARAM NUMBER IS GARBAGE.

PF/PFX	0300	192	UNDEFINED PARAMETER.
PF/PFX	0310	200	ILLEGAL LFN.
PF/PFX	0320	208	BINARY ZERO OR NON-ALFA IN LFN.
PF/PFX	0330	816	NO OR ILLEGAL PFN.
PF	0340	224	UNDEFINED PARAMETER.
PFX	0350	232	BINARY ZERO 1ST PARAM.
PFX	0360	240	BINARY ZERO PARAMETER.
PFX	0370	248	UNDEFINED PARAMETER.
PF/PFX	0400	256	NEITHER LFN NOR PFN.
PF/PFX	0500	320	ALFA IN REAL TIME SWITCH.
PF/PFX	0510	328	NON-ALFA, NON-NUMERIC IN REAL TIME SWITCH.
LF	0600	384	BINARY ZERO/BLANKS IN DISPOSITION CODE.
LF	0610	392	ILLEGAL DISPOSITION CODE.
LF	0700	448	FID BINARY ZERO/BLANKS.
LF	1000	512	TID BINARY ZERO/BLANKS.
LF	1100	576	BINARY ZERO OR BLANKS IN DISPOSITION CODE EQUIP.
LF	1110	584	ILLEGAL PARAMETER.
LF	1200	640	BINARY ZERO OR BLANKS IN FORMS CODE.
	1300	704	ALFA OR BINARY ZERO IN DIGIT STRING.
	1310	712	SPECIAL CHARACTER IN DIGIT STRING.
PF/LF	1320	720	STRING TOO LONG.
PF/LF	1330	728	TOO MANY PARAMETERS.
PF/LF	1340	736	MORE THAN 5 PASSWORDS.
PF/LF	1350	744	FAILED ABSOLUTE CHAR TEST.
PF/LF	1360	752	EXCEEDED BINARY MAXIMUM.
	1400	816	XX IN TAPEXX > 99.
	5000	2560	USER HAS CLOBBED CORE.
	6001	3073	INVALID TID.

NOS/BE-GENERATED MESSAGES FOR PF/PFX *

DEC	OCT	COMND	MEANING
0	000	ALL	FUNCTION SUCCESSFUL
1	001		PFN/ID ERROR
2	002	A,P	LFN ALREADY IN USE
3	003	CEPR	UNKNOWN LFN
4	004	C	TOO MANY CYCLES (5 MAX)
5	005	C,E	PF CATALOG FULL
6	006		NO LFN OR PFN
8	010	C,E	LATEST INDEX NOT WRITTEN
9	011	C	FILE NOT ON A PF DEVICE
10	012	A	FILE NOT CATALOGED, SN=<SETNAME>
11	013	A	ARCHIVE RETRIEVAL ABORTED
12	014	C,R	BAD LPF COMMUNICATION
13	015	C	CY LIMIT REACHED (999 MAX)
14	016	C	PF DIRECTORY FULL
15	017	CEPR	FUNCTION ATTEMPTED ON A NON-PERMANENT FILE
16	020		FCN ATTEMPTED ON NON-LOCAL FILE
17	021	A	IMPROPER ARCHIVE RETRIEVAL CALL
18	022	C	FILE NEVER ASSIGN TO A DEVICE
19	023	A	CYCLE INCOMPLETE OR DUMPED
20	024	A	FILE ALREADY ATTACHED
21	025	A	FILE ARCHIVED
22	026		ILLEGAL CHARACTER IN FDB PARAM
23	027		ILLEGAL LFN
24	030	A	FILE DUMPED
25	031		ILLEGAL FUNCTION CODE
26	032	P	PURGE ATTEMPT IGNORED; USE RB PARAMETER
27	033		ALTER NEEDS EXCLUSIVE ACCESS
28	034		FDB IS TOO LARGE
29	035	C	FILE ALREADY IN SYSTEM
30	036	A	NO APF SPACE
31	037		PERMISSION CONFLICTS
32	040		ILLEGAL SETNAME SPECIFIED
33	041		DEVICE NOT MOUNTED AT CTL POINT
34	042		RBT CHAIN TOO LARGE FOR PFC
35	043	A,P	FILE RESIDES ON UNAVAILABLE DEVICE
36	044	A,P	FILE NOT AVAILABLE
56	070		PFM STOPPED BY SYSTEM
** 57	071		INCORRECT PERMISSION
** 58	072		FILE DEFINITION BLOCK ADDRESS INVALID (NOT RETURNED TO FDB)
** 59	073		I/O ERROR ON PFD/PFC READ/WRITE

* - THE TEXT CAN BE OBTAINED IN A PROGRAM BY USING SUBROUTINE
PFRF IN LIBRARY NSRDC5 OR NSRDC.

** - ALWAYS CAUSES ABNORMAL JOB TERMINATION

NOS/BE-GENERATED MESSAGES FOR LF(ROUTE) *

DEC	OCT	MEANING
1	001	INVALID LFN - DSP
2	002	CANNOT ROUTE NON-ALLOCATABLE EQUIPMENT
3	003	CANNOT ROUTE PERMANENT FILE
4	004	NO PERMISSION TO ROUTE THIS FILE
5	005	ROUTE TO INPUT NOT IMMEDIATE - IGNORED
6	006	IMMEDIATE ROUTING - NO FILE - IGNORED
7	007	INVALID DISPOSITION CODE - ROUTING IGNORED
8	010	INVALID FID - ROUTING IGNORED
9	011	DSP ABORTED BY SYSTEM
10	012	DSP PARAMETER OUTSIDE FL
11	013	PRIORITY SPECIFICATION IGNORED
12	014	E1200 SPECIFIED - INTERCOM USED (DSP)
13	015	E1200 SPECIFIED - INTERCOM USED (DSP)
14	016	CANNOT ROUTE INPUT FILE
15	017	DSP COMPLETE BIT ALREADY SET
16	020	FILE ON DISMOUNTABLE DEVICE - ROUTING IGNORED
17	021	TID NOT ALPHANUMERIC - ROUTING IGNORED
18	022	FORMS CODE NOT ALPHANUMERIC - ROUTING IGNORED
19	023	INVALID LINK TYPE - ROUTING IGNORED (DSP)
20	024	FILE NOT ON QUEUE DEVICE - ROUTING IGNORED
21	025	PRE-DAYFILE LFN AND NO DC=IN - ROUTE IGNORED
22	026	PRE-DAYFILE FILE NOT FOUND - ROUTE IGNORED

* - THE TEXT CAN BE OBTAINED IN A PROGRAM BY USING SUBROUTINE
ROUTERC IN LIBRARY NSRDC5 OR NSRDC.

SUBROUTINE AND FUNCTION SUBPROGRAMS REQUIRED
PART OF LANGUAGE
NONE
OTHERS
NONE

AUTHORS
J. REM
DTNSRDC CODE 1892.3

DATE WRITTEN: 12/21/78

DATE(S) REVISED
02/15/83 - MAKE AC 10 CHARACTERS
- MAKE TID 3 CHARACTERS
- ADD DEF AS SYNONYM FOR DEFER
- ADD PARAMETER "DAY" TO CONTROL DAYFILE MESSAGES
- INSTALL IN SYSTEM LIBRARY SYSLIB
05/12/83 - MAKE TAPE NUMBER 3 DIGITS
- ALLOW TO RUN IN A CAPSULE

LOCATION OF DECKS
SOURCE
CODE 1892.3
OBJECT
PART OF OPERATING SYSTEM (MODULE NAME PRMFIL)

ANOTHER COPY OF THIS DOCUMENT
BEGIN,DOCGET,,NSRDC,,PF,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'PFR'

PURPOSE

SUPPLY DESCRIPTION OF PERMANENT FILE FUNCTION RETURN CODE

FUNCTIONAL CATEGORIES: Q0

LANGUAGE: FORTRAN IV EXTENDED

COMPUTERS (OPERATING SYSTEMS)

CDC 6000/CYBER 170 (NOS/BE)

REMARKS

THE DESCRIPTIONS ARE THOSE FOUND IN THE "NOS/BE VERSION 1
REFERENCE MANUAL" (60493800 H) ON PAGE 83.

USAGE

CALL PFR (IRC, A)

DESCRIPTION OF PARAMETERS

IRC - RETURN CODE FROM THE PERMANENT FILE FUNCTION

A - 5-WORD ARRAY WHICH WILL CONTAIN THE DESCRIPTION OF THE
SUPPLIED 'IRC'
(IF 'IRC' IS INVALID, 'UNKNOWN RETURN CODE' IS
RETURNED)

CM REQUIRED: 1075B

SUBPROGRAMS REQUIRED

PART OF LANGUAGE

MOVEIT

OTHERS

NONE

AUTHOR

DAVID V SOMMER - DTNSRDC CODE 1892.2

DATE WRITTEN: 05/18/76

DATE(S) REVISED

02/14/77 - UPDATE FOR NOS/BE 1.0

07/15/80 - UPDATE FOR NOS/BE 1.4 (LEVEL 508)

03/21/83 - RECOMPILE AT NOS/BE LEVEL 552

LOCATION OF DECKS

SOURCE

UPDATE LIBRARY ON MSS: NSRDCPL,UN=CSYS

OBJECT

EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC,,PFR,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'PLOTPR'

PURPOSE

PRODUCE PRINTER PLOTS WHICH MAY HAVE:

FUNCTIONAL CATEGORIES: J5

LANGUAGE: FORTRAN IV

COMPUTERS (OPERATING SYSTEMS)

CDC 6000/CYBER 170 (NOS/BE)

REMARKS

A CALL TO 'INITPLO' WILL SET THE DEFAULT VALUES.

THE MINIMUM SIZE OF A GRID IS 101 X 101 POINTS (THIS IS 1-1/2 COMPUTER PAGES). IF MORE THAN 101 VALUES FOR THE INDEPENDENT VARIABLE ARE GIVEN, THE REQUIRED INTEGRAL NUMBER OF 100-POINT GRIDS ARE AUTOMATICALLY JOINED TOGETHER.

THE NAME AND VALUES OF THE INDEPENDENT VARIABLE (AND X TITLE) ARE GIVEN IN THE LEFT MARGIN. THE NAMES, SCALES AND PLOTTING CHARACTERS (A-I) FOR THE DEPENDENT VARIABLES ARE GIVEN AT THE TOP OF THE PAGE WITH THE PAGE TITLE AND Y TITLE ABOVE THEM.

- 1) ANY NUMBER OF PLOTS PER RUN
- 2) ANY NUMBER OF VALUES FOR THE INDEPENDENT VARIABLE
- 3) UP TO 9 DEPENDENT VARIABLES PER PLOT.

USAGE

COMMON /PLO/ NRUN, NPLOT, ITP(6), ITY(6), ITX(6),
NMPAG, MAXSCA, SCA(10), FROM(10)

...

CALL INITPLO

C SET ANY SPECIAL VALUES IN COMMON /PLO/ AFTER 'CALL INITPLO'

...

C WRITE DATA FOR THE PLOT

DO 5 I=1,NOPTS

5 WRITE (NFILE) VARIND(I), VARDEP1(I), ..., VARDEPN(I)

...

CALL PLOTPR (NFILE, NUMVAR, IVAR)

DESCRIPTION OF PARAMETERS

NFILE - FORTRAN LOGICAL UNIT NUMBER OF FILE CONTAINING
THE DATA VALUES, INDEPENDENT FOLLOWED BY DEPENDENT

NUMVAR - NUMBER OF VARIABLES (UP TO 10)
(TOTAL: INDEPENDENT + DEPENDENT)

IVAR - 10-WORD ARRAY WITH ALPHANUMERIC NAMES FOR THE
VARIABLES WHICH WILL APPEAR ON THE PLOT

ADDITIONAL INFORMATION IS PROVIDED THRU LABELLED COMMON
BLOCK /PLO/

NRUN - NUMBER OF THIS RUN (DEFAULT: 1)
NPLOT - NUMBER OF PLOT (DEFAULT: 1)

ITP - PAGE TITLE (DEFAULT: BLANK)
 ITY - Y TITLE (DEFAULT: BLANK)
 ITX - X TITLE (DEFAULT: BLANK)
 (TITLE ARRAYS ARE 6 WORDS EACH OF UP TO 6
 CHARACTERS PER WORD - 6A6 FORMAT)
 NUMPAG - NUMBER OF DOUBLE PAGES TO SPREAD THE PLOT
 OVER (NO MORE THAN 100 POINTS PER PAGE)
 (DEFAULT: 1)
 MAXSCA - SCALING OPTION
 1 - OPTIMUM SCALING IS CALCULATED FOR EACH
 VARIABLE (DEFAULT)
 2 - PLOT ALL DEPENDENT VARIABLES ON THE
 SAME SCALE
 (IF THE PROGRAMMER SCALES ANY OF THE
 DEPENDENT VARIABLES, THIS OPTION IS DEFAULTED)
 SCA AND FROM -
 ARRAYS CONTAINING THE INCREMENTS AND THE
 STARTING VALUES FOR EACH VARIABLE.
 IF ONE OF THESE ARRAYS IS USED FOR A VARIABLE,
 BOTH MUST BE USED.
 IF THERE ARE MORE THAN 101 VALUES FOR THE
 INDEPENDENT VARIABLE, THOSE VALUES MUST HAVE A
 CONSTANT INCREMENT AND THE SCALING IS ALWAYS
 BASED ON THAT INCREMENT.
 (DEFAULT: OPTIMUM SCALE AND STARTING VALUE
 ARE CALCULATED FOR EACH VARIABLE)

REMEMBER TO PUT 'TAPE_NFILE' INTO PROGRAM STATEMENT OF THE
 MAIN PROGRAM.

CM REQUIRED: 1514B

SUBPROGRAMS REQUIRED

PART OF LANGUAGE

ABS	ALOG10	AMAX1	AMIN1	AND
COMPL	EOF	OR	REWIND	SHIFT

OTHERS

DRAWGD
 INITGD
 INITPLO

AUTHOR

ADAPTED FROM MIMIC BY ANN BANDURSKI - DTNSRDC CODE 1833

DATE WRITTEN: 05/22/72

DATE(S) REVISED

03/23/83 - RECOMPILE AT NOS/BE LEVEL 552

LOCATION OF DECKS

SOURCE

UPDATE LIBRARY ON MSS: NSRDCPM, UN=CSYS (*DECK AMPLOTP)

OBJECT

EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT

BEGIN, DOCGET, , NSRDC, , PLOTPR, OUTPUT, MSACCES=<PASSWORD>.

SUBROUTINE 'POLYN'

PURPOSE

LEAST SQUARES POLYNOMIAL FIT

FUNCTIONAL CATEGORIES: E2

LANGUAGE: FORTRAN IV

COMPUTERS (OPERATING SYSTEMS)

CDC 6000/CYBER 170 (NOS/BE)

REMARKS

FIT AN N-TH DEGREE POLYNOMIAL TO SETS OF POINTS (X(I), Y(I), Z(I), ...), WHERE X IS THE INDEPENDENT VARIABLE IN EACH CASE, (I=1,2,...,N).

$$PN(X) = A(0) + A(1)*X + A(2)*X**2 + \dots + A(N)*X**N$$

USAGE

CALL POLYN (ND, NP, NC, X, Y, NAPT, WORKA, V, SUM)

DESCRIPTION OF PARAMETERS

- ND - DEGREE OF POLYNOMIAL (N)
- NP - NUMBER OF POINTS IN SET OF OBSERVATIONS
(X(I), Y(I), Z(I), ...)
- NC - NUMBER OF CURVES TO BE FITTED
(E.G., Y, Z, ...)
- X - ARRAY CONTAINING THE INDEPENDENT VARIABLE
- Y - ARRAY CONTAINING THE DEPENDENT VARIABLE(S)
MUST BE DIMENSIONED AT LEAST NP TIMES NC.
Y(1), Y(2), ... MUST BE CONTIGUOUS IN MEMORY.
Z(1) NEED NOT FOLLOW Y(N) IMMEDIATELY.
- NAPT - NUMBER OF LOCATIONS BETWEEN SETS OF DATA
Y, Z, ... (NUMBER OF WORDS BETWEEN Y(1) AND Z(1).) ALL SETS Y, Z, ... MUST BE EQUALLY SPACED.
- WORKA - WORK ARRAY USED IN MATRIX SOLUTION OF THE (ND+1) SETS OF LINEAR EQUATIONS. MUST BE DIMENSIONED AT LEAST (ND+1)**2.
- V - OUTPUT ARRAY USED IN MATRIX SOLUTION FOR VECTOR.
MUST BE DIMENSIONED AT LEAST (ND+1) TIMES NC.
V(1), ..., V(ND+1) WILL CONTAIN COEFFICIENTS A(0), ..., A(N) OF THE FIRST CURVE.
- SUM - WORK ARRAY FOR SUMS OF POWERS OF X.
MUST BE DIMENSIONED AT LEAST (2*ND+1).

CM REQUIRED: 321B

METHOD

LEAST SQUARES - MINIMIZING SUM OF SQUARES OF DEVIATIONS FROM AVERAGE.

SUBPROGRAMS REQUIRED

PART OF LANGUAGE

NONE

PART OF PROGRAM

ENXEN

OTHERS

NONE

AUTHOR

J. N. BROOKS (SHARE ROUTINE NUMBER 848)

DATE WRITTEN: 01/29/60

DATE(S) REVISED

09/23/83 - RECOMPILE AT NOS/BE LEVEL 552

LOCATION OF DECKS

SOURCE

UPDATE LIBRARY ON MSS: NSRDCPM,UN=CSYS (*DECK ARPLN1)

OBJECT

EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC,,POLYN,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'PROOT'

PURPOSE

FIND ALL ROOTS OF A REAL POLYNOMIAL

FUNCTIONAL CATEGORIES: C2 B4

LANGUAGE: FORTRAN IV

COMPUTERS (OPERATING SYSTEMS)

CDC 6000/CYBER 170 (NOS/BE)

REMARKS

THE POLYNOMIAL HAS THE FORM:

$$A_1 + A_2 X + \dots + A_{N+1} X^{**N} = 0$$

USAGE

CALL PROOT (N, A, U, V, H, B, C, CONV, NPLUS2)

DESCRIPTION OF PARAMETERS

- N - DEGREE OF THE POLYNOMIAL TO BE SOLVED
- A - ARRAY (DIMENSIONED N+2) CONTAINING THE COEFFICIENTS
IN THE ORDER INDICATED ABOVE
- U - ARRAY (DIMENSIONED N+2) WHICH WILL CONTAIN THE
REAL PARTS OF THE ROOTS
- V - ARRAY (DIMENSIONED N+2) WHICH WILL CONTAIN THE
IMAGINARY PARTS OF THE ROOTS
- H,B,C - WORK ARRAYS (EACH DIMENSIONED N+2)
- CONV - CONVERGENCE CRITERION. INITIALLY SET BY PROOT TO
1.0E-35 (FAR BELOW THE ACTUAL STARTING CONVERGENCE
CRITERION OF 5.0E-20 (CDC). IF THE POLYNOMIAL HAS
NOT CONVERGED AFTER A PRESCRIBED NUMBER OF TRIES,
THE CONVERGENCE CRITERION IS RELAXED. IF, UPON EXIT
FROM PROOT, CONV IS NOT 1.0E-35, THE CONVERGENCE
CRITERION HAS BEEN RELAXED TO THE NUMBER GIVEN.
- NPLUS2 - MUST BE SET TO N+2

CM REQUIRED: 421B

METHOD

THE ROUTINE CONVERGES SIMULTANEOUSLY TOWARD A LINEAR FACTOR
AND A QUADRATIC FACTOR BY NEWTON'S AND BAIRSTOW'S METHODS,
RESPECTIVELY. WHEN A ROOT IS FOUND BY ONE METHOD, ITERATION
CONTINUES WITH BOTH METHODS USING THEIR MOST RECENT GUESSES.

SUBPROGRAMS REQUIRED

PART OF LANGUAGE

ABS SIGN Sqrt

OTHERS

NONE

AUTHORS

MIRIAM SHAPIRO
HARVEY ABRAMSON - NEW YORK UNIVERSITY

DATE WRITTEN: UNKNOWN - ADAPTED FROM LOS ALAMOS ROUTINE
LA-PROOT BY T. L. VORDAN (MS)

DATE(S) REVISED

11/65 - CONVERTED TO CDC 6600 (HA)
09/20/83 - RECOMPILE AT NOS/BE LEVEL 552

LOCATION OF DECKS

SOURCE

UPDATE LIBRARY ON MSS: NSRDCPM,UN=CSYS

OBJECT

EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC,,PROOT ,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'PRTFL'

PURPOSE

PRINT CURRENT FL (OR PUT INTO DAYFILE)

FUNCTIONAL CATEGORIES: Q0 J2

LANGUAGE: FORTRAN IV

COMPUTERS (OPERATING SYSTEMS)

CDC 6000/CYBER 170 (NOS/BE)

REMARKS

NONE

USAGE

CALL PRTFL (IOUT)

DESCRIPTION OF PARAMETER

IOUT - FORTRAN LOGICAL UNIT NUMBER

(0=PUT INTO DAYFILE; N=WRITE ON TAPEN)

CM REQUIRED: 50B

OUTPUT UNITS

UNIT	#	LFN	USE
------	---	-----	-----

USER SPECIFIES...		LISTABLE OUTPUT	
-------------------	--	-----------------	--

SUBPROGRAMS REQUIRED

PART OF LANGUAGE

REMARK

OTHERS

FTNRFL - GET CURRENT FL

AUTHOR

DAVID V SOMMER - DTNSRDC CODE 1892.2

DATE WRITTEN: 04/16/75

DATE(S) REVISED

02/15/83 - RECOMPILE AT NOS/BE LEVEL 552

LOCATION OF DECKS

SOURCE

UPDATE LIBRARY ON MSS: NSRDCPL,UN=CSYS

OBJECT

EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC,,PRTFL,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'PRTIME'

PURPOSE

GET AND PRINT CPA, CPB, CP, PP, IO AND WALL CLOCK TIMES
SINCE LAST CALL AND PRINT USER-SUPPLIED MESSAGE

FUNCTIONAL CATEGORIES: Q4 J4 NO

LANGUAGE: FORTRAN IV

COMPUTERS (OPERATING SYSTEMS)
CDC 6000/CYBER 170 (NOS/BE)

REMARKS
NONE

USAGE

CALL PRTIME (IOUNIT, TIMES, MSG)
CALL PRTIME (IOUNIT, TIMES, 0)

DESCRIPTION OF PARAMETERS

IOUNIT - OUTPUT UNIT FOR PRINTED LINE
(EITHER FORTRAN LOGICAL UNIT NUMBER (1-99) OR
1- TO 7-CHARACTER LOCAL FILE NAME, LEFT-ADJ,
ZERO-FILLED (E.G., 6LOUTPUT))

TIMES - 7-WORD ARRAY TO CONTAIN THE FOLLOWING:
1 - ELAPSED CPA TIME IN SECONDS
2 - ELAPSED CPB TIME IN SECONDS (0.0)
3 - ELAPSED CP TIME IN SECONDS (CPA+CPB=CPA)
4 - ELAPSED PP TIME IN SECONDS
5 - ELAPSED IO TIME IN SECONDS
6 - ELAPSED WALL CLOCK TIME (HH.MM.SS.)
7 - ELAPSED WALL CLOCK TIME IN SECONDS

MSG - 5-WORD MESSAGE TO BE PRINTED
(IF SUPPLIED AS HOLLERITH CONSTANT, MAY BE FEWER
THAN 5 WORDS. SEE EXAMPLE BELOW)
(IF MSG(1) IS 0 (OR 1L0 OR 1H0), HEADINGS, BUT NOT
TIMES, WILL BE PRINTED.)

CM REQUIRED: 102B

OUTPUT UNITS

UNIT #	LFN	USE
-----	-----	-----
USER SPECIFIES...	LISTABLE OUTPUT	

EXAMPLE

```
      PROGRAM TEST (OUTPUT=128, .....  
      REAL TIMES(7)  
C     GET INITIAL TIMES AND PRINT HEADING  
      CALL PRTIME (6LOUTPUT, TIMES, 0)  
      .....  
C     GET ELAPSED TIMES AND PRINT WITH MESSAGE  
      CALL PRTIME (6LOUTPUT, TIMES, "TEST NUMBER 1")  
      ...  
C     NEW HEADINGS ARE NOT NEEDED, SO CALL ELTIME DIRECTLY  
      CALL ELTIME (TIMES)  
      .....  
C     GET ELAPSED TIMES AND PRINT WITH MESSAGE  
      CALL PRTIME (6LOUTPUT, TIMES, "TEST NUMBER 2")  
      ...  
      END
```

SUBPROGRAMS REQUIRED

PART OF LANGUAGE

NONE

OTHERS

ELTIME - GET ELAPSED TIME SINCE LAST CALL

FINDCHR - FIND FIRST OCCURRENCE OF CHARACTER IN ARRAY

AUTHOR

DAVID V SOMMER - DTNSRDC CODE 1892.2

DATE WRITTEN: 04/20/76

DATE(S) REVISED

03/23/83 - RECOMPILE AT NOS/BE LEVEL 552

LOCATION OF DECKS

SOURCE

UPDATE LIBRARY ON MSS: NSRDCPL,UN=CSYS

OBJECT

EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC,,PRTIME,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'PUTCHA'
FUNCTION 'PUTCHA'

PURPOSE

INSERT CHARACTER INTO SPECIFIED POSITION IN AN ARRAY

FUNCTIONAL CATEGORIES: M4

LANGUAGE: FORTRAN IV EXTENDED

COMPUTERS (OPERATING SYSTEMS)
CDC 6000/CYBER 170 (NOS/BE)

REMARKS

'PUTCHA' IS AN ENTRY POINT IN 'GETCHA'.

USAGE

CALL PUTCHA (A, N, CH)
VARIABLE = PUTCHA (A, N, CH)

DESCRIPTION OF PARAMETERS

A - ARRAY INTO WHICH CHARACTER IS TO BE INSERTED
N - POSITION AT WHICH CHARACTER IS TO BE INSERTED
(POSITION 1 IS LEFT-MOST 6-BIT CHARACTER IN A(1))
CH - CHARACTER TO BE INSERTED (IN 1R FORMAT)
(WHEN USED AS A FUNCTION, PUTCHA WILL CONTAIN THE WORD
IN 'A' WHICH WAS CHANGED)

CM REQUIRED: 56B (INCLUDES GETCHA)

SUBPROGRAMS REQUIRED

PART OF LANGUAGE
SHIFT
OTHERS
NONE

AUTHOR

DAVID V SOMMER - DTNSRDC CODE 1892.2

DATE WRITTEN: 03/16/76

DATE(S) REVISED

03/23/83 - RECOMPILE AT NOS/BE LEVEL 552

LOCATION OF DECKS

SOURCE
UPDATE LIBRARY ON MSS: NSRDCPL,UN=CSYS (*DECK GETCHA)
OBJECT
EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC,,PUTCHA,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'PUTCHR'
FUNCTION 'PUTCHR'

PURPOSE

INSERT CHARACTER INTO SPECIFIED POSITION IN A WORD

FUNCTIONAL CATEGORIES: M4

LANGUAGE: FORTRAN IV EXTENDED

COMPUTERS (OPERATING SYSTEMS)
CDC 6000/CYBER 170 (NOS/BE)

REMARKS

'PUTCHR' IS AN ENTRY POINT IN 'GETCHR'.

USAGE

CALL PUTCHR (A, N, CH)
VARIABLE = PUTCHR (A, N, CH)

DESCRIPTION OF PARAMETERS

A - WORD INTO WHICH CHARACTER IS TO BE INSERTED
N - POSITION AT WHICH CHARACTER IS TO BE INSERTED
(POSITION 1 IS LEFT-MOST 6-BIT CHARACTER IN A)
CH - CHARACTER TO BE INSERTED (IN 1H FORMAT)
(WHEN USED AS A FUNCTION, PUTCHR WILL CONTAIN THE SAME
AS 'A')

CM REQUIRED: 46B (INCLUDES GETCHR)

SUBPROGRAMS REQUIRED

PART OF LANGUAGE
SHIFT
OTHERS
NONE

AUTHOR

FROM BIMED PACKAGE

DATE WRITTEN:

DATE(S) REVISED

1975 - DAVID V SOMMER - DTNSRDC CODE 1892.2
03/23/83 - RECOMPILE AT NOS/BE LEVEL 552

LOCATION OF DECKS

SOURCE
UPDATE LIBRARY ON MSS: NSRDCPL,UN=CSYS (*DECK GETCHR)
OBJECT
EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC,,PUTCHR,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'QSORT'

PURPOSE

IN-CORE ASCENDING SORT FOR REAL ARRAYS LARGER THAN 500 WORDS

FUNCTIONAL CATEGORIES: M1

LANGUAGE: FORTRAN IV

COMPUTERS (OPERATING SYSTEMS)

CDC 6000/CYBER 170 (NOS/BE)

REMARKS

'QSORT' IS THE MOST EFFICIENT SORT AVAILABLE (AS OF DATE BELOW) FOR THE SORTING IN CORE OF ARRAYS LARGER THAN 500 WORDS.

THIS ROUTINE IS A TRANSLATION OF ALGORITHM 402, COMM. ACM, NOV, 1970.

IF THE JOB ABORTS WITH THE MESSAGE "ABORT IN QSORT WITH MN=<MN>", CHECK IF MN EXCEEDS KL (CURRENTLY KL=46). IF SO, THE VALUE OF KL AND THE DIMENSION OF ARRAY K MUST BE SET HIGHER (TRY DOUBLING IT).

ON THE CDC:

WRITE A DUMMY SUBROUTINE TO SET KL AND THE DIMENSION OF K GREATER.

THIS SUBROUTINE MIGHT HAVE THE FORM:

```
SUBROUTINE DUMMY
COMMON /QSORT/ KL, K(<NEW>)
KL = <NEW>
RETURN
END
```

A CALL TO THIS SUBROUTINE MUST OCCUR BEFORE ANY CALL TO QSORT; THE BEST PLACE BEING ONE OF THE FIRST STATEMENTS IN THE MAIN PROGRAM.

USAGE

CALL QSORT (A, I)

DESCRIPTION OF PARAMETERS

A - REAL ARRAY TO BE SORTED INTO ASCENDING ORDER

I - NUMBER OF WORDS IN 'A' TO BE SORTED

CM REQUIRED: 207B (+ 57B COMMON)

ERROR MESSAGE

ABORT IN QSORT WITH MN=<MN>
SEE REMARKS.

SUBPROGRAMS REQUIRED

PART OF LANGUAGE

DISPLA (CDC)

OTHERS

ZABORT - NON-EXISTENT ROUTINE TO FORCE ABORT

AUTHORS

C FLINK - KPS NWL

DTNSRDC CODE 1892

DATE WRITTEN: 11/25/70 - CF

DATE(S) REVISED

01/30/81 - DVS - ADD DAYFILE ERROR MESSAGE

- CHANGE ABORT PROCESS

02/17/81 - DVS - CONVERT TO B7700

03/23/83 - RECOMPILE AT NOS/BE LEVEL 552

LOCATION OF DECKS

SOURCE

UPDATE LIBRARY ON MSS: NSRDCPL,UN=CSYS

OBJECT

EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC,,QSORT,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'QSORT1'

PURPOSE

IN-CORE ASCENDING SORT WITH RE-ORDERING OF ASSOCIATED ARRAY
(FOR REAL ARRAYS LARGER THAN 500 WORDS)

FUNCTIONAL CATEGORIES: M1

LANGUAGE: FORTRAN IV

COMPUTERS (OPERATING SYSTEMS)
CDC 6000/CYBER 170 (NOS/BE)

REMARKS

'QSORT1' IS THE MOST EFFICIENT SORT AVAILABLE (AS OF DATE
BELOW) FOR THE SORTING IN CORE OF ARRAYS LARGER THAN 500
WORDS.

THIS ROUTINE IS A TRANSLATION OF ALGORITHM 402, COMM. ACM
NOV, 1970.

IF THE ARRAY 'T' IS NOT NEEDED, USE 'QSORT'.

IF THE JOB ABORTS WITH THE MESSAGE "ABORT IN QSORT1 WITH
MN=<MN>", CHECK IF MN EXCEEDS KL (CURRENTLY KL=46).
IF SO, THE VALUE OF KL AND THE DIMENSION OF ARRAY K MUST BE
SET HIGHER (TRY DOUBLING IT).

ON THE CDC:

WRITE A DUMMY SUBROUTINE TO SET KL AND THE DIMENSION OF K
GREATER.

THIS SUBROUTINE MIGHT HAVE THE FORM:

```
SUBROUTINE DUMMY
COMMON /QSORT/ KL, K(<NEW>)
KL = <NEW>
RETURN
END
```

A CALL TO THIS SUBROUTINE MUST OCCUR BEFORE ANY CALL TO
QSORT1; THE BEST PLACE BEING ONE OF THE FIRST STATEMENTS
IN THE MAIN PROGRAM.

USAGE

CALL QSORT1 (A, I, T)

DESCRIPTION OF PARAMETERS

A - REAL ARRAY TO BE SORTED INTO ASCENDING ORDER
I - NUMBER OF WORDS IN 'A' TO BE SORTED
T - ASSOCIATED ARRAY TO BE REORDERED

CM REQUIRED: 271B (+ 57B COMMON)

SUBPROGRAMS REQUIRED

PART OF LANGUAGE

DISPLA (CDC)

OTHERS

ZABORT - NON-EXISTENT ROUTINE TO FORCE ABORT

AUTHORS

C FLINK - KPS NWL

DTNSRDC CODE 1892

DATE WRITTEN: 11/30/70

DATE(S) REVISED

01/30/81 - DVS - ADD DAYFILE ERROR MESSAGE
- CHANGE ABORT PROCESS

02/17/81 - DVS - CONVERT TO B7700

03/23/83 - RECOMPILE AT NOS/BE LEVEL 552

LOCATION OF DECKS

SOURCE

UPDATE LIBRARY ON MSS: NSRDCPL,UN=CSYS

OBJECT

EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC,,QSORT1,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'QUADG'

PURPOSE

INTEGRAL BY GAUSS-LEGENDRE 10-POINT QUADRATURE

FUNCTIONAL CATEGORIES: D1

LANGUAGE: FORTRAN IV EXTENDED

COMPUTERS (OPERATING SYSTEMS)

CDC 6000/CYBER 170 (NOS/BE)

REMARKS

APPROXIMATES

$$\text{INTEGRAL } F(X) \text{ } DX = \frac{XU-XL}{2} * \text{SUM} [W_I * F\left(\frac{Z_I(XU-XL)+XU+XL}{2}\right)]$$

WHERE W_I ARE WEIGHT FACTORS

Z_I ARE ROOTS OF LEGENDRE POLYNOMIAL

INTEGRAL IS FROM XL TO XU.

USAGE

CALL QUADG (XL, XU, FNC, Y)

DESCRIPTION OF PARAMETERS

XL - LOWER LIMIT OF INTEGRATION

XU - UPPER LIMIT OF INTEGRATION

FNC - THE EXTERNAL FUNCTION FOR EVALUATING THE INTEGRAND F(X).

(USER MUST SUPPLY THE FUNCTION 'FNC' WITH ONE ARGUMENT FOR EVALUATING F(X), THE INTEGRAND. FNC MUST BE DECLARED EXTERNAL IN THE ROUTINE CALLING QUADG.)

Y - THE RESULTING INTEGRAL VALUE

CM REQUIRED: 153B

METHOD

LET A = .5*(XU+XL)

B = XU - XL,

THEN, SINCE THE Z_I'S ARE SYMMETRIC ABOUT ZERO,

$$Y = B * \text{SUM-FROM-0-TO-4} ((W_I/2) * (FNC(A + (Z_I/2)*B) + FNC(A - (Z_I/2)*B)))$$

REFERENCE

"APPLIED NUMERICAL METHODS" BY B. CARNAHAN, H. LUTHER AND J. WILKES, WILEY, 1969, P. 103.

SUBPROGRAMS REQUIRED
PART OF LANGUAGE
NONE
OTHERS
NONE

AUTHOR
SUSAN VOIGT - DTNSRDC CODE 1892

DATE WRITTEN: 09/71

DATE(S) REVISED
09/20/83 - RECOMPILE AT NOS/BE LEVEL 552

LOCATION OF DECKS
SOURCE
UPDATE LIBRARY ON MSS: NSRDCPM,UN=CSYS (*DECK AMQUADG)
OBJECT
EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT
BEGIN,DOCGET,,NSRDC,,QUADG,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'RCPA'

PURPOSE

READ (A PORTION OF) CONTROL POINT AREA

FUNCTIONAL CATEGORIES: K2

LANGUAGE: CDC 6000 CP COMPASS

COMPUTERS (OPERATING SYSTEMS)

CDC 6000/CYBER 170 (NOS/BE)

REMARKS

NONE

USAGE

CALL RCPA (ISTART, NWORDS, AREA)

DESCRIPTION OF PARAMETERS

ISTART - STARTING WORD IN CONTROL POINT AREA

NWORDS - NUMBER OF WORDS TO READ

AREA - ARRAY TO HOLD THE SPECIFIED WORDS
(AREA(2) THRU AREA(NWORDS+1))

CM REQUIRED: 43B

SUBPROGRAMS REQUIRED

PART OF LANGUAGE

NONE

OTHERS

NONE

AUTHOR

MIKE GOLDEN - DTNSRDC CODE 1844

DAVID V SOMMER - DTNSRDC CODE 1892.2

DATE WRITTEN: 11/75

DATE(S) REVISED

12/03/75

03/22/83 - CHANGE 60-BIT ADDRESSES TO 18 BITS TO MAKE IT
USABLE IN A CAPSULE

- ADD COMMENT FOR LOAD MAP LIST

LOCATION OF DECKS

SOURCE

UPDATE LIBRARY ON MSS: NSRDCPL,UN=CSYS

OBJECT

EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC,,RCPA,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'RECOVRD'

PURPOSE

ON RECOVERY, PRINT EXCHANGE JUMP PACKAGE, RA+0 THRU RA+77B
AND ENDRUN INDICATOR

FUNCTIONAL CATEGORIES: N2

LANGUAGE: FORTRAN IV EXTENDED

COMPUTERS (OPERATING SYSTEMS)
CDC 6000/CYBER 170 (NOS/BE)

REMARKS
NONE

USAGE

...
EXTERNAL RECOVRD
...
CALL RECOVR (RECOVRD, 77B, 0)
...

--OR--

...
EXTERNAL ANY
...
CALL RECOVR (ANY, 77B, 0)
...
SUBROUTINE ANY (EXCHJP, ENDRUN, RAO)
DIMENSION EXCHJP(17)
CALL RECOVRD (EXCHJP, ENDRUN, RAO)
...

DESCRIPTION OF PARAMETERS

EXCHJP - 17-WORD ARRAY TO HOLD EXCHANGE JUMP PACKAGE
ENDRUN - ENDRUN INDICATOR (WILL HAVE MEANING ONLY IF SECOND
FORM OF USAGE IS USED AND IF ENDRUN IS SET BEFORE
THE CALL TO RECOVRD)
RAO - RA+0 POINTER (NOT USED BY THIS SUBROUTINE)

CM REQUIRED: 606B

OUTPUT UNITS

LFN	USE
OUTPUT	LISTABLE OUTPUT

SUBPROGRAMS REQUIRED

PART OF LANGUAGE
SHIFT
OTHERS
GETRA - GET RA+0 THRU RA+77B

AUTHOR

DAVID V SOMMER - DTNSRDC CODE 1892.2

DATE WRITTEN: 06/19/74

DATE(S) REVISED

03/23/83 - RECOMPILE AT NOS/BE LEVEL 552

LOCATION OF DECKS

SOURCE

UPDATE LIBRARY ON MSS: NSRDCPL,UN=CSYS

OBJECT

EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC,,RECOVRD,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'REDUCE'

PURPOSE

REDUCE FL TO MINIMUM OR REQUEST ADDITIONAL FL RELATIVE TO
START OF BLANK COMMON

FUNCTIONAL CATEGORIES: Q0

LANGUAGE: FORTRAN IV EXTENDED

COMPUTERS (OPERATING SYSTEMS)
CDC 6000/CYBER 170 (NOS/BE)

REMARKS

NOT RECOMMENDED. USE COMMON MEMORY MANAGER CALLS.

REQUIRES FTN4,STATIC OR FTN5,STATIC.

USAGE

CALL REDUCE - REDUCE TO FIRST WORD OF BLANK COMMON
CALL REDUCE (I) - ADJUST TO 'I' WORDS AFTER START OF BLANK
COMMON

DESCRIPTION OF PARAMETER

I - IF PRESENT, NUMBER OF WORDS PAST START OF BLANK COMMON

CM REQUIRED: 36B (PLUS 1 IN BLANK COMMON)

SUBPROGRAMS REQUIRED

PART OF LANGUAGE
LOCF SHIFT
OTHERS

MFETCH - GET SPECIFIED WORD IN USER'S FL
MSET - SET SPECIFIED WORD IN USER'S FL

AUTHOR

? - NWL

DATE WRITTEN: ?

DATE(S) REVISED

03/23/83 - RECOMPILE AT NOS/BE LEVEL 552

LOCATION OF DECKS

SOURCE
UPDATE LIBRARY ON MSS: NSRDCPL,UN=CSYS
OBJECT
EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET.,NSRDC,,REDUCE,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'REPLAC'

PURPOSE

REPLACE ONE CHARACTER BY ANOTHER IN AN ARRAY

FUNCTIONAL CATEGORIES: M4

LANGUAGE: FORTRAN IV EXTENDED

COMPUTERS (OPERATING SYSTEMS)

CDC 6000/CYBER 170 (NOS/BE)

REMARKS

ALL PARAMETERS ARE TYPE 'INTEGER'

USAGE

CALL REPLAC (A, NA, OLD, NEW)

DESCRIPTION OF PARAMETERS

A - ARRAY TO BE PROCESSED

NA - NUMBER OF WORDS IN 'A' TO BE PROCESSED

OLD - OLD CHARACTER (1R FORMAT)

NEW - NEW CHARACTER (1R FORMAT)

CM REQUIRED: 24B

SUBPROGRAMS REQUIRED

PART OF LANGUAGE

SHIFT

OTHERS

NONE

ARITHMETIC STATEMENT FUNCTIONS

L91FMT - FAST L-FORMAT DECODE (LEFT-ADJ, ZERO-FILLED)

R110FMT - FAST R-FORMAT DECODE (RIGHT-ADJ, ZERO-FILLED)

AUTHOR

DAVID V SOMMER - DTNSRDC CODE 1892.2

DATE WRITTEN: 1973

DATE(S) REVISED

03/21/83 - RECOMPILE AT NOS/BE LEVEL 552

LOCATION OF DECKS

SOURCE

UPDATE LIBRARY ON MSS: NSRDCPL,UN=CSYS

OBJECT

EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC,,REPLAC,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'REPLACM'

PURPOSE

REPLACE OLD CHARACTERS WITH NEW CHARACTERS

FUNCTIONAL CATEGORIES: M4

LANGUAGE: FORTRAN IV EXTENDED

COMPUTERS (OPERATING SYSTEMS)

CDC 6000/CYBER 170 (NOS/BE)

REMARKS

ALL PARAMETERS ARE TYPE 'INTEGER'

USAGE

CALL REPLACM (A, NA, OLD, NEW, NCH)

DESCRIPTION OF PARAMETERS

A - ARRAY TO BE PROCESSED

NA - NUMBER OF WORDS IN 'A' TO BE PROCESSED

OLD - ARRAY OF OLD CHARACTERS (1R FORMAT)

NEW - ARRAY OF CORRESPONDING NEW CHARACTERS (1R FORMAT)

NCH - NUMBER OF CHANGE PAIRS (DIMENSION OF 'OLD' AND 'NEW')

CM REQUIRED: 44B

SUBPROGRAMS REQUIRED

PART OF LANGUAGE

SHIFT

OTHERS

NONE

ARITHMETIC STATEMENT FUNCTIONS

L91FMT - FAST L-FORMAT DECODE (LEFT-ADJ, ZERO-FILLED)

R110FMT - FAST R-FORMAT DECODE (RIGHT-ADJ, ZERO-FILLED)

AUTHOR

DAVID V SOMMER - DTNSRDC CODE 1892.2

DATE WRITTEN: 05/21/75

DATE(S) REVISED

03/23/83 - RECOMPILE AT NOS/BE LEVEL 552

LOCATION OF DECKS

SOURCE

UPDATE LIBRARY ON MSS: NSRDCPL,UN=CSYS

OBJECT

EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC,,REPLACM,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'REPLHI'

PURPOSE

REPLACE ALL CHARACTERS GREATER THAN SPECIFIED CHARACTER WITH
NEW CHARACTER

FUNCTIONAL CATEGORIES: M4

LANGUAGE: FORTRAN IV EXTENDED

COMPUTERS (OPERATING SYSTEMS)
CDC 6000/CYBER 170 (NOS/BE)

REMARKS

ALL PARAMETERS ARE TYPE 'INTEGER'

USAGE

CALL REPLHI (A, NA, OLD, NEW)

DESCRIPTION OF PARAMETERS

A - ARRAY TO BE PROCESSED
NA - NUMBER OF WORDS IN 'A' TO BE PROCESSED
OLD - OLD CHARACTER (1R FORMAT)
NEW - NEW CHARACTER (1R FORMAT)

CM REQUIRED: 25B

SUBPROGRAMS REQUIRED

PART OF LANGUAGE
SHIFT
OTHERS
NONE

ARITHMETIC STATEMENT FUNCTIONS

L91FMT - FAST L-FORMAT DECODE (LEFT-ADJ, ZERO-FILLED)
R110FMT - FAST R-FORMAT DECODE (RIGHT-ADJ, ZERO-FILLED)

AUTHOR

DAVID V SOMMER - DTNSRDC CODE 1892.2

DATE WRITTEN: 01/26/76

DATE(S) REVISED

03/23/83 - RECOMPILE AT NOS/BE LEVEL 552

LOCATION OF DECKS

SOURCE
UPDATE 'LIBRARY ON MSS: NSRDCPL,UN=CSYS
OBJECT
EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC,,REPLHI,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'REPLLO'

PURPOSE

REPLACE ALL CHARACTERS LESS THAN SPECIFIED CHARACTER WITH
NEW CHARACTER

FUNCTIONAL CATEGORIES: M4

LANGUAGE: FORTRAN IV EXTENDED

COMPUTERS (OPERATING SYSTEMS)
CDC 6000/CYBER 170 (NOS/BE)

REMARKS

ALL PARAMETERS ARE TYPE 'INTEGER'

USAGE

CALL REPLLO (A, NA, OLD, NEW)

DESCRIPTION OF PARAMETERS

A - ARRAY TO BE PROCESSED
NA - NUMBER OF WORDS IN 'A' TO BE PROCESSED
OLD - OLD CHARACTER (1R FORMAT)
NEW - NEW CHARACTER (1R FORMAT)

CM REQUIRED: 25B

SUBPROGRAMS REQUIRED

PART OF LANGUAGE
SHIFT
OTHERS
NONE

ARITHMETIC STATEMENT FUNCTIONS

L91FMT - FAST L-FORMAT DECODE (LEFT-ADJ, ZERO-FILLED)
R110FMT - FAST R-FORMAT DECODE (RIGHT-ADJ, ZERO-FILLED)

AUTHOR

DAVID V SOMMER - DTNSRDC CODE 1892.2

DATE WRITTEN: 01/26/76

DATE(S) REVISED

03/23/83 - RECOMPILE AT NOS/BE LEVEL 552

LOCATION OF DECKS

SOURCE
UPDATE LIBRARY ON MSS: NSRDCPL,UN=CSYS
OBJECT
EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC,,REPLLO,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'REPLNE'

PURPOSE

REPLACE ALL CHARACTERS (EXCEPT SPECIFIED CHARACTER) WITH A
SPECIFIED CHARACTER

FUNCTIONAL CATEGORIES: M4

LANGUAGE: FORTRAN IV EXTENDED

COMPUTERS (OPERATING SYSTEMS)
CDC 6000/CYBER 170 (NOS/BE)

REMARKS

ALL PARAMETERS ARE TYPE 'INTEGER'

USAGE

CALL REPLNE (A, NA, OLD, NEW)

DESCRIPTION OF PARAMETERS

A - ARRAY TO BE PROCESSED
NA - NUMBER OF WORDS IN 'A' TO BE PROCESSED
OLD - OLD CHARACTER (1R FORMAT)
NEW - NEW CHARACTER (1R FORMAT)

CM REQUIRED: 24B

SUBPROGRAMS REQUIRED

PART OF LANGUAGE
SHIFT
OTHERS
NONE

ARITHMETIC STATEMENT FUNCTIONS

L91FMT - FAST L-FORMAT DECODE (LEFT-ADJ, ZERO-FILLED)
R110FMT - FAST R-FORMAT DECODE (RIGHT-ADJ, ZERO-FILLED)

AUTHOR

DAVID V SOMMER - DTNSTDC CODE 1892.2

DATE(S) REVISED

03/21/83 - RECOMPILE AT NOS/BE LEVEL 552

LOCATION OF DECKS

SOURCE
UPDATE LIBRARY ON MSS: NSRDCPL,UN=CSYS
OBJECT
EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC,,REPLNE.OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'REQTBL'

PURPOSE

READ THE INTERCOM USER TABLE

FUNCTIONAL CATEGORIES: QO

LANGUAGE: CDC CP COMPASS

COMPUTERS (OPERATING SYSTEMS)

CDC 6000/CYBER 170 (NOS/BE)

REMARKS

THIS DOCUMENT DESCRIBES ONLY ONE FUNCTION OF THE SUBROUTINE.

USAGE

CALL REQTBL (CMDORD, BUFFER, DUMMY1, LBUFR, NWORDS, DUMMY2,
ERR)

DESCRIPTION OF PARAMETERS

CMDORD - COMMAND ORDINAL (MUST BE 1)

BUFFER - ARRAY TO RECEIVE THE TABLE
(NORMALLY 16 WORDS)

DUMMY1 - NOT USED (SHOULD BE 0)

LBUFR - LENGTH OF BUFFER ARRAY

NWORDS - NUMBER OF WORDS OF TABLE DESIRED
(UP TO LBUFR WORDS WILL BE TRANSFERRED)

DUMMY2 - NOT USED (MAY BE OMITTED IF ERROR CODE NOT NEEDED)

ERR - ERROR RETURN CODE (OPTIONAL)
0 - NO ERROR

CM REQUIRED: 33B

SUBPROGRAMS REQUIRED

PART OF LANGUAGE

NONE

OTHERS

TBL - PP PROGRAM TO PROCESS THE REQUEST

AUTHOR

VERNON GREENHILL - DTNSRDC CODE 1892.3 (CDC)

DATE WRITTEN: 10/05/84

DATE(S) REVISED

LOCATION OF DECKS

SOURCE DECK

UPDATE LIBRARY ON MSS: NSRDCPL,UN=CSYS

OBJECT DECK

EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC,,REQTBL,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'REQUEST'

PURPOSE

CALLABLE REQUEST FUNCTION

FUNCTIONAL CATEGORIES: Q3

LANGUAGE: FORTRAN IV EXTENDED

COMPUTERS (OPERATING SYSTEMS)
CDC 6000/CYBER 170 (NOS/BE)

REMARKS

NO LONGER SUPPORTED SINCE SUBROUTINE 'REQUEST' HAS BEEN REPLACED BY SYSTEM SUBROUTINE 'LF' (SEE SUBROUTINE 'PF' DOCUMENT).

ASSIGNMENT OF EQUIPMENT MAY BE REQUESTED FROM A RUNNING CENTRAL PROCESSOR PROGRAM BY THE REQUEST SUBROUTINE, WHICH HAS THE EFFECT OF A REQUEST CARD.

USAGE

CALL REQUEST (IRC, LFN, ICODE, SN)
CALL REQUEST (IRC, LFN, ICODE)
CALL REQUEST (IRC, LFN)

DESCRIPTION OF PARAMETERS

IRC - OUTPUT: RIGHT-JUSTIFIED NOS/BE-GENERATED ERROR RETURN CODE
IRC=0 - REQUEST WAS SUCCESSFUL

LFN - CONTENTS DETERMINED BY ICODE
IF ICODE IS NON-ZERO, LFN IS A 1-7 CHARACTER LOCAL FILE NAME, LEFT-JUSTIFIED, ZERO- OR BLANK-FILLED (E.G., 5LTAPE7).
IF ICODE IS ZERO (OR MISSING), LFN IS AN ARRAY CONSTRUCTED AS DESCRIBED IN NOS/BE REFERENCE MANUAL, PAGE 12-23 ON.

ICODE - DETERMINES CONTENTS OF LFN AND EFFECT OF REQUEST
ICODE 0 OR MISSING - LFN IS AN ARRAY CONTAINING PARAMETERS FOR REQUEST MACRO
ICODE = "*Q", 2H*Q OR 2L*Q - LFN IS 1-7 CHARACTER LOCAL FILE NAME AND REQUEST HAS EFFECT OF REQUEST,LFN,*Q.
ICODE ANYTHING ELSE - LFN IS 1-7 CHARACTER LOCAL FILE NAME AND REQUEST HAS THE EFFECT OF REQUEST,LFN,*PF.

SN - OPTIONAL SN (*PF ONLY)
WHEN USED, IS 1-7 CHARACTER USER DEVICE SET NAME (HAS EFFECT OF REQUEST,LFN,*PF,SN=SETNAME.)

CM REQUIRED: 121B

EXAMPLES

REQUEST,TAPE1,*PF. BECOMES
CALL REQUEST (IRC, 5LTAPE1, 1)

REQUEST,TAPE2,*Q. BECOMES
CALL REQUEST (IRC, 5LTAPE2, "*Q")

REQUEST,TAPE3,*PF,SN=MYSET1. BECOMES
CALL REQUEST (IRC, "TAPE3", "*PF", "MYSET1")

SUBPROGRAMS REQUIRED

PART OF LANGUAGE

LOCF SHIFT

OTHERS

IZONK NUMVAR ZPFMAC

AUTHORS

JAMES BLACK, MIKE CHERNICK - DTNSRDC CODE 1832
DAVID V SOMMER - DTNSRDC CODE 1892.2

DATE WRITTEN: 05/26/71

DATE(S) REVISED

01/10/75 - V3.5 - MC
01/27/77 - DVS - ADD *Q
03/24/77 - DVS - ADD SN
03/21/83 - RECOMPILE AT NOS/BE LEVEL 552

LOCATION OF DECKS

SOURCE

UPDATE LIBRARY ON MSS: NSRDCPL,UN=CSYS

OBJECT

EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC,,REQUEST,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'RFFT'

PURPOSE

FAST FOURIER TRANSFORM OF A REAL TABULATED FUNCTION

FUNCTIONAL CATEGORIES: E2

LANGUAGE: FORTRAN IV EXTENDED

COMPUTERS (OPERATING SYSTEMS)

CDC 6000/CYBER 170 (NOS/BE)

REMARKS

NONE

USAGE

CALL RFFT (A, M, INV, S, IFERR)

DESCRIPTION OF PARAMETERS

- A - THE ARRAY CONTAINING A REAL TABULATED ONE-DIMENSIONAL FUNCTION. 'A' MUST BE DIMENSIONED AS A POWER OF 2 AND REQUIRES 4 ADDITIONAL LOCATIONS BEYOND THE LENGTH OF THE DATA. TOTAL DIMENSION FOR 'A' IS $2^{**}(M+1)+4$.
ON OUTPUT 'A' CONTAINS THE FOURIER TRANSFORM. A(1) AND A(2) CONTAIN, RESPECTIVELY, THE REAL AND IMAGINARY ZERO-CYCLE COMPONENTS; A(3) AND A(4) CONTAIN THE FUNDAMENTAL FREQUENCY COMPONENTS, ETC.
- M - ONE LESS THAN THE SMALLEST INTEGER BASE 2 LOGARITHM THAT HAS AN ANTILOG WHICH WILL CONTAIN ALL THE ELEMENTS TO BE TRANSFORMED. FOR EXAMPLE, IF THE ARRAY TO BE TRANSFORMED CONTAINS 28 POINTS, M MUST BE SET TO 4.
- INV - SCRATCH ARRAY REQUIRING 1/8 THE DIMENSION OF 'A'
- S - SCRATCH ARRAY REQUIRING 1/8 THE DIMENSION OF 'A'
- IFERR - ERROR RETURN CODE
= 0 -- NORMAL COMPLETION
<>0 -- ERRORS IN SUBROUTINE ARGUMENTS

NOTE: $3 \leq M \leq 20$. THIS IS BASED ON AN ARRAY WHICH HAS A LENGTH THAT CAN BE EXPRESSED AS A POWER OF 2. IF THE DATA OCCUPIES LESS SPACE THAN $2^{**}(M+1)$, THE REMAINING LOCATIONS MUST BE SET TO ZERO OR ANOTHER APPROPRIATE CONSTANT.

CM REQUIRED: 250B

METHOD

THIS OPERATION MAKES USE OF THE SEPARABLE PROPERTIES OF THE FOURIER COEFFICIENTS OF THE REAL AND IMAGINARY COMPONENTS OF THE COMPLEX VECTOR. THIS IS ALMOST A SPECIAL CASE OF THE DUAL USE OF THE COOLEY-TUKEY ALGORITHM DESCRIBED IN REFERENCE 2. REFERENCES TO THIS METHOD CAN BE FOUND IN REFERENCE 3 ALSO.

IN BRIEF, A SCALED VERSION OF THE FIRST PORTION OF THE REAL ARRAY IS PLACED IN THE REAL COMPONENT OF THE VECTOR, WHILE A SCALED VERSION OF THE SECOND PORTION OF THE ARRAY IS PLACED IN THE COMPLEX COMPONENT. THE ALGORITHM IS PERFORMED IN NORMAL FASHION ON THE COMPLEX ARRAY. THE COEFFICIENTS FOR THE REAL ARRAY ARE OBTAINED BY PROPERLY COMBINING AND REORDERING THE FOURIER COEFFICIENTS FROM THE COMPLEX PROCESSING.

REFERENCES

1. COOLEY, J. W. AND TUKEY, J. W. "AN ALGORITHM FOR THE MACHINE CALCULATION OF COMPLEX FOURIER SERIES," MATH. COMPUT. 19, 90 (APRIL 1965), 297-301.
2. GODFREY, M. D., BINGHAM, C., AND TUKEY, J. W., "MODERN TECHNIQUES OF POWER SPECTRUM ESTIMATION," IEEE TRANS. ON AUDIO AND ELECTROACOUSTICS (JUNE 1967), PP. 56-66.
3. SINGLETON, RICHARD C., "ON COMPUTING THE FAST FOURIER TRANSFORM," COMM. OF THE ACM, VOL 10, NO 10, OCTOBER 1967.
4. SYSTEM/360 SCIENTIFIC SUBROUTINE PACKAGE, IBM TECHNICAL PUBLICATIONS DEPT., 1967.

SUBPROGRAMS REQUIRED

PART OF LANGUAGE

COS FLOAT SIN

OTHERS

FFT - FAST FOURIER TRANSFORM OF A COMPLEX TAB FCN

AUTHORS

WES RICE
DUANE HARDER
LOS ALAMOS SCIENTIFIC LABORATORY

VIM ROUTINE LASL C330A

DATE WRITTEN: 07/24/68

DATE(S) REVISED

02/69 - DH

09/20/83 - RECOMPILE AT NOS/BE LEVEL 552

LOCATION OF DECKS

SOURCE

UPDATE LIBRARY ON MSS: NSRDCPM,UN=CSYS

OBJECT

EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC,,RFFT,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'RFSN'

PURPOSE

INVERSE FAST FOURIER TRANSFORM

FUNCTIONAL CATEGORIES: E2

LANGUAGE: FORTRAN IV EXTENDED

COMPUTERS (OPERATING SYSTEMS)

CDC 6000/CYBER 170 (NOS/BE)

REMARKS

NONE

USAGE

CALL RFSN (A, M, INV, S, IFERR)

DESCRIPTION OF PARAMETERS

- A - THE ARRAY CONTAINING THE REAL AND IMAGINARY FOURIER COEFFICIENT. A(1) AND A(2) CONTAIN, RESPECTIVELY, THE REAL AND IMAGINARY COMPONENTS, ETC. $2^{**}(M+1)+1$ AND $2^{**}(M+1)+2$ ARE THE FINAL FREQUENCY SUBSCRIPTS WHICH ARE USED IN THE SYNTHESIS. 'A' MUST BE DIMENSIONED AT LEAST $2^{**}(M+1)+4$.
ON OUTPUT 'A' CONTAINS THE INVERSE FOURIER TRANSFORM.
- M - ONE LESS THAN THE SMALLEST INTEGER BASE 2 LOGARITHM THAT HAS AN ANTILOG WHICH WILL CONTAIN ALL THE ELEMENTS TO BE TRANSFORMED. FOR EXAMPLE, IF THE ARRAY TO BE TRANSFORMED CONTAINS 28 POINTS, M IS SET TO 4. THIS RESULT WOULD REQUIRE 17 PAIRS OF COEFFICIENTS.
- INV - SCRATCH ARRAY REQUIRING 1/8 THE DIMENSION OF 'A'
- S - SCRATCH ARRAY REQUIRING 1/8 THE DIMENSION OF 'A'
- IFERR - ERROR RETURN CODE
 - = 0 -- NORMAL COMPLETION
 - <>0 -- ERRORS IN SUBROUTINE ARGUMENTS

NOTE: $3 < M < 20$. ALL COEFFICIENTS MUST BE DEFINED;
THEREFORE ALL $2^{**}(M+1)$ REAL AND IMAGINARY COEFFICIENTS
MUST BE SET TO APPROPRIATE VALUES.

CM REQUIRED: 254B

METHOD

THIS OPERATION MAKES USE OF THE SEPARABLE PROPERTIES OF THE FOURIER COEFFICIENTS OF THE REAL AND IMAGINARY COMPONENTS OF A COMPLEX VECTOR. THE ALGORITHM IS ACCOMPLISHED BY PERFORMING IN REVERSE ORDER THE INVERSE OF EACH STEP IN SUBROUTINE RFFT. THIS IS ALMOST A SPECIAL CASE OF THE DUAL USE OF THE COOLEY-TUKEY ALGORITHM DESCRIBED IN REFERENCE 2. ANOTHER SIMILAR TECHNIQUE IS DESCRIBED IN REFERENCE 3.

REFERENCES

1. COOLEY, J. W. AND TUKEY, J. W. "AN ALGORITHM FOR THE MACHINE CALCULATION OF COMPLEX FOURIER SERIES," MATH. COMPUT. VOL 19, NO 90 (APRIL 1965), 297-301.
2. GODFREY, M. D., BINGHAM, C., AND TUKEY, J. W., "MODERN TECHNIQUES OF POWER SPECTRUM ESTIMATION," IEEE TRANS. ON AUDIO AND ELECTROACOUSTICS (JUNE 1967), PP. 56-66.
3. SINGLETON, RICHARD C., "ON COMPUTING THE FAST FOURIER TRANSFORM," COMM. OF THE ACM, VOL 10, NO 10, OCTOBER 1967.
4. SYSTEM/360 SCIENTIFIC SUBROUTINE PACKAGE, IBM TECHNICAL PUBLICATIONS DEPT., 1967.

SUBPROGRAMS REQUIRED

PART OF LANGUAGE

COS SIN

OTHERS

FFT - FAST FOURIER TRANSFORM OF A COMPLEX TAB FCN

AUTHORS

WES RICE

DUANE HARDER

LOS ALAMOS SCIENTIFIC LABORATORY

VIM ROUTINE LASL C331A

DATE WRITTEN: 08/07/68

DATE(S) REVISED

02/69 - DH

09/20/83 - RECOMPILE AT NOS/BE LEVEL 552

LOCATION OF DECKS

SOURCE

UPDATE LIBRARY ON MSS: NSRDCPM,UN=CSYS

OBJECT

EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC,,RFSN,OUTPUT,MSACCES=<PASSWORD>.

FUNCTION 'RNDMIZ'

PURPOSE

EMULATE BASIC LANGUAGE 'RANDOMIZE' STATEMENT (CAN BE USED TO
GUARANTEE FIRST CALL TO RANF WILL RESULT IN A DIFFERENT
NUMBER WITH EACH EXECUTION OF A PROGRAM)

FUNCTIONAL CATEGORIES: V1

LANGUAGE: FORTRAN IV EXTENDED

COMPUTERS (OPERATING SYSTEMS)
CDC 6000/CYBER 170 (NOS/BE)

REMARKS
NONE

USAGE
R = RNDMIZ (N)

DESCRIPTION OF PARAMETERS

N - DUMMY ARGUMENT - IGNORED
RNDMIZ - WILL RETURN A RANDOM NUMBER SIMILAR TO THAT
OBTAINED BY RANF

CM REQUIRED: 23B

METHOD

THE RANF SEED IS CHANGED USING THE CURRENT CP TIME
(FRACTIONAL PART ONLY)

SUBPROGRAMS REQUIRED

PART OF LANGUAGE
INT RANF SECOND
OTHERS
NONE

AUTHOR

DAVID V SOMMER - DTNSRDC CODE 1892.2

DATE WRITTEN: 11/08/77

DATE(S) REVISED

03/23/83 - RECOMPILE AT NOS/BE LEVEL 552

LOCATION OF DECKS

SOURCE
UPDATE LIBRARY ON MSS: NSRDCPL,UN=CSYS
OBJECT
EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT

BEGIN.DOCGET,,NSRDC,,RNDMIZ,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'ROUTE'

PURPOSE

CALLABLE ROUTE COMMAND

FUNCTIONAL CATEGORIES: Q0

LANGUAGE: FORTRAN IV EXTENDED

COMPUTERS (OPERATING SYSTEMS)

CDC 6000/CYBER 170 (NOS/BE)

REMARKS

NO LONGER SUPPORTED SINCE SUBROUTINE 'ROUTE' HAS BEEN REPLACED BY SYSTEM SUBROUTINE 'LT' (SEE SUBROUTINE 'PF' DOCUMENT).

THE FILE TO BE ROUTED MUST BE ON A QUEUE DEVICE.

THE CALLING PROGRAM MUST CLOSE THE FILE BEFORE 'ROUTE' IS CALLED. AN FTM SEQUENTIAL FILE (WRITE, PRINT, PUNCH) MAY BE "CLOSED" BY ISSUING A 'REWIND N' BEFORE THE CALL TO 'ROUTE'. IF THE FILE IS NOT CLOSED, THE FINAL BUFFER MAY NOT BE ROUTED.

USAGE

CALL ROUTE (IRC, IPRMS, NW)

DESCRIPTION OF PARAMETERS

IRC - ERROR RETURN CODE

NOS/BE-GENERATED

DEC	OCT	MEANING
0	000	FUNCTION SUCCESSFUL
1	001	INVALID LFN - DSP
2	002	CANNOT ROUTE NON-ALLOCATABLE EQUIPMENT
3	003	CANNOT ROUTE PERMANENT FILE
4	004	NO PERMISSION TO ROUTE THIS FILE
5	005	ROUTE TO INPUT NOT IMMEDIATE - IGNORED
6	006	IMMEDIATE ROUTING - NO FILE - IGNORED
7	007	INVALID DISPOSITION CODE - ROUTING IGNORED
8	010	INVALID FID - ROUTING IGNORED
9	011	DSP ABORTED BY SYSTEM
10	012	DSP PARAMETER OUTSIDE FL
11	013	PRIORITY SPECIFICATION IGNORED
12	014	RMT ROUTING, NO ID - CENTRAL SITE ASSUMED
13	015	E1200 SPECIFIED - INTERCOM USED (DSP)
14	016	CANNOT ROUTE INPUT FILE
15	017	DSP COMPLETE BIT ALREADY SET
16	020	FILE ON DISMOUNTABLE DEVICE - ROUTING IGNORED
17	021	TID NOT ALPHANUMERIC - ROUTING IGNORED
18	022	FORMS CODE NOT ALPHANUMERIC - ROUTING IGNORED
19	023	INVALID LINK TYPE - ROUTING IGNORED (DSP)
20	024	FILE NOT ON QUEUE DEVICE - ROUTING IGNORED
21	025	PRE-DAYFILE LFN AND NO DC=IN - ROUTE IGNORED
22	026	PRE-DAYFILE FILE NOT FOUND - ROUTE IGNORED

AD-A148 792

COMPUTER CENTER CDC LIBRARIES/NSRD (SUBPROGRAMS)(U)
DAVID W TAYLOR NAVAL SHIP RESEARCH AND DEVELOPMENT
CENTER BET. D V SOMMER ET AL. JUN 84

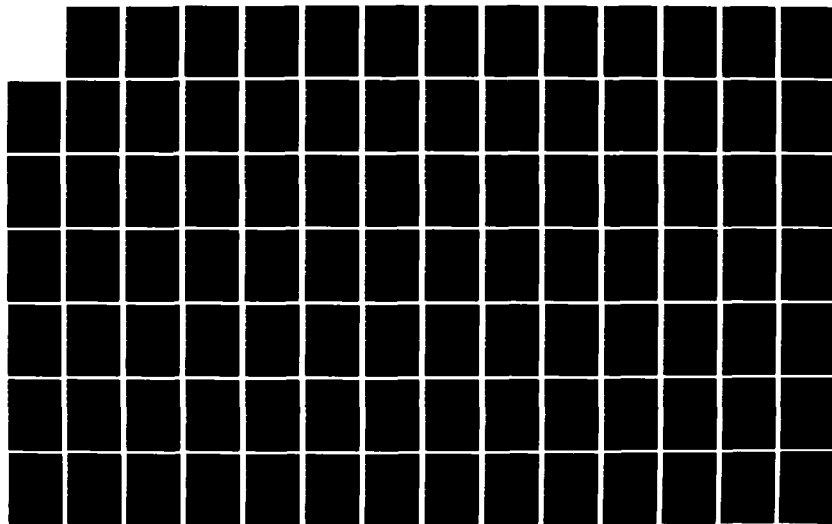
4/5

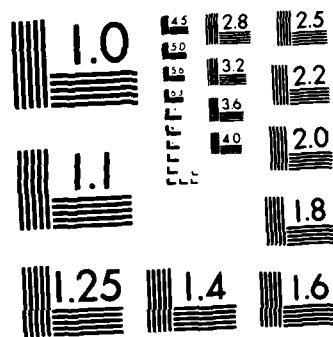
UNCLASSIFIED

DTNSRDC/CMLD-84-12

F/G 9/2

NL





MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A

IPRMS - PARAMETERS FOR ROUTE
(UNUSED FIELDS MUST BE SET TO ZERO)

IPRMS	CONTENTS	FORMAT
1	LFN	1-7 CHAR, LEFT**
2	DC	0 FOR DEFAULT -OR- 2-CHAR DISPOSITION CODE, LEFT**
3	TID	0 -OR- 1LC - ROUTE TO CENTRAL SITE -OR- 2-CHAR TERMINAL ID, LEFT** -OR- 4LHERE - ROUTE TO THIS TERMINAL
4	FID	1-7 CHAR FILE ID -OR- 1L* -OR- 1-5 CHAR FILE ID, PRECEDED BY * (ALL LEFT**)
5	DEF	0 -OR- 3LDEF - TO DEFER ROUTING UNTIL END-OF-JOB
6		NON-ZERO TO RETURN THE JOB NAME IN THIS WORD
7	FC	0 -OR- 2-CHAR FORMS CODE, LEFT**
8	EC	0 - USE DEFAULT FOR PRINT: 2LB4, 2LB6, 2LA6, 2LA9 FOR PUNCH: 2LSB, 5L80COL, 3L026, 3L029, 5LASCII
9	IC	ONE OF: 0 OR 3LDIS - DISPLAY CODE 5LASCII - ASCII 3LBIN - BINARY
10	STID	3-CHAR STATION (SITE) ID, LEFT**
11	PRI	PRIORITY FOR INTERACTIVELY ROUTED OUTPUT FILE BEING ROUTED TO THE ROUTING TERMINAL - 1-4 DIGIT OCTAL VALUE (0000B-7777B)
12	REP	FOR ALL OTHER FILES - 0 REPEAT COUNT (0-31 (37B))
13	NCD	0 -OR- 1 - NO COMPLEMENTARY DAYFILE (VALID ONLY IF IPRMS(5)=3LDEF)
14	SC	SPACING CODE FOR 580 PRINTER (0-77B; 0 IS DEFAULT ARRAY) (NOT AVAILABLE YET AT DTNSRDC)

** LEFT=LEFT-JUSTIFIED, BLANK OR ZERO PADDED

NW - NUMBER OF LAST ELEMENT IN IPRMS
(IF OMITTED, NW=13)

CM REQUIRED: 401B

EXAMPLES

- 1) ASSUME THE PROGRAM HAS WRITTEN FILE 'TAPE7' TO BE PRINTED AT CENTRAL SITE:

```

...
INTEGER IPRMS(14)
...
IPRMS(1) = 5LTAPE7
IPRMS(2) = 2LPR
IPRMS(3) = 1LC
IPRMS(4) = 1L*
...
REWIND 7
CALL ROUTE (IRC, IPRMS, 4)
...

```

THIS WILL SIMULATE: ROUTE,TAPE7,DC=PR,TID=C,FID=*.

- 2) A PROGRAM WISHES TO PUNCH FILE 'PUNCH' AT REMOTE TERMINAL '011' AT END OF JOB:

```

...
INTEGER IPRMS(14)
...
IPRMS(1) = 5LPUNCH
IPRMS(2) = 2LPU
IPRMS(3) = 3L011
IPRMS(4) = 1L*
IPRMS(5) = 3LDEF
IPRMS(6) = 1
...
CALL ROUTE (IRC, IPRMS, 6)
IF (IRC .EQ. 0) PRINT 1, IPRMS(6)
1 FORMAT (" TAPE7 WILL BE PRINTED WITH JOB NAME " A7)
...

```

THIS WILL SIMULATE: ROUTE,PUNCH,DC=PU,TID=011,FID=*,DEF.

- 3) A PROGRAM CREATES A 'JOB' ON FILE 'TAPE99' TO BE SUBMITTED TO THE SAME INPUT QUEUE AS THE CREATING JOB:

```

...
INTEGER IPRMS(14)
...
IPRMS(1) = 6LTAPE99
IPRMS(2) = 2LIN
IPRMS(3) = 4LHERE
...
WRITE (99, 1)
99 FORMAT ("JOB CARD" / "CHARGE CARD" / "....")
REWIND 99
CALL ROUTE (IRC, IPRMS, 3)

```

THIS WILL SIMULATE: ROUTE,TAPE99,DC=IN,TID.

SUBPROGRAMS REQUIRED

PART OF LANGUAGE

AND LOCF MAX0 MIN0 OR
SHIFT

OTHERS

BZFILL - CHANGE BLANKS TO 00B
HERE - GET TERMINAL ID FOR THIS JOB
HEX3 - CONVERT 3-DIGIT HEX TO 2-CHAR
MOVECM - MOVE AN ARRAY
TRAILBZ - CHANGE TRAILING BLANKS TO 00B
ZSYSEQ - CALL THE SYSTEM

ARITHMETIC STATEMENT FUNCTIONS

FAST L-FORMAT DECODE (LEFT-ADJ, ZERO-FILLED)

L11FMT L21FMT L31FMT L52FMT L71FMT

FAST R-FORMAT DECODE (RIGHT-ADJ, ZERO-FILLED)

R18FMT R21FMT

AUTHOR

DAVID V SOMMER - DTNSRDC CODE 1892.2

DATE WRITTEN: 12/08/75

DATE(S) REVISED

01/24/77 - ADD REP PARAMETER, CHANGE PRI DESCRIPTION
11/30/77 - ADD NCD PARAMETER
10/01/78 - CHANGE TO 3-CHARACTER TID
02/24/81 - UPGRADE FOR LEVEL 508
04/04/83 - ADD 14-TH PARAMETER SC FOR FUTURE USE
03/23/83 - RECOMPILE AT NOS/BE LEVEL 552

LOCATION OF DECKS

SOURCE

UPDATE LIBRARY ON MSS: NSRDCPL,UN=CSYS

OBJECT

EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC,,ROUTE,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'ROUTERC'

PURPOSE

SUPPLY DESCRIPTION OF ROUTE RETURN CODE

FUNCTIONAL CATEGORIES: Q0

LANGUAGE: FORTRAN IV EXTENDED

COMPUTERS (OPERATING SYSTEMS)

CDC 6000/CYBER 170 (NOS/BE)

REMARKS

THE DESCRIPTIONS ARE THOSE FOUND IN THE "NOS/BE VERSION 1
REFERENCE MANUAL" (60493800 H) ON PAGE 7-82.

USAGE

CALL ROUTERC (IRC, A)

DESCRIPTION OF PARAMETERS

IRC - RETURN CODE FROM SUBROUTINE 'ROUTE'

A - 5-WORD ARRAY WHICH WILL CONTAIN THE DESCRIPTION OF THE
SUPPLIED 'IRC'
(IF 'IRC' IS INVALID, 'UNKNOWN RETURN CODE' IS
RETURNED)

CM REQUIRED: 625B

SUBPROGRAMS REQUIRED

PART OF LANGUAGE

NONE

OTHERS

MOVEIT - MOVE AN ARRAY

AUTHOR

DAVID V SOMMER - DTNSRDC CODE 1892.2

DATE WRITTEN: 12/15/77

DATE(S) REVISED

03/23/83 - RECOMPILE AT NOS/BE LEVEL 552

LOCATION OF DECKS

SOURCE

UPDATE LIBRARY ON MSS: NSRDCPL,UN=CSYS

OBJECT

EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC,,ROUTERC,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'SBYT'
FUNCTION 'SBYT'

PURPOSE
STORE VARIABLE LENGTH BYTE

FUNCTIONAL CATEGORIES: M4

LANGUAGE: CDC 6000 CP COMPASS

COMPUTERS (OPERATING SYSTEMS)
CDC 6000/CYBER 170 (NOS/BE)

REMARKS
STORES A 1 TO 60-BIT BYTE FROM ONE WORD INTO ANY POSITION IN
A SECOND WORD WITHOUT DISTURBING THE REMAINING PART OF THAT
WORD.

USAGE
CALL SBYT (N, LENGTH, INTO, FROM)
-OR-
VARIABLE = SBYT (N, LENGTH, INTO, FROM)

DESCRIPTION OF PARAMETERS

N - BEGINNING BIT POSITION IN WORD <INTO> WHERE THE
BYTE WILL BE PLACED. BITS ARE NUMBERED FROM 1 TO
60 FROM RIGHT TO LEFT.
LENGTH - LENGTH OF THE BYTE IN BITS. THIS LENGTH STARTS
WITH THE RIGHTMOST BIT OF <FROM>.
INTO - WORD INTO WHICH THE BYTE WILL BE PLACED.
FROM - WORD FROM WHICH THE BYTE WILL BE TAKEN FROM THE
LOW ORDER BITS.

NOTE: IN THE SECOND FORM, <VARIABLE> AND <INTO> WILL
CONTAIN THE SAME VALUE. THUS, THEY MAY HAVE THE
SAME VARIABLE NAME.

NOTE: BITS 1 THRU <LENGTH> OF WORD <FROM> ARE PLACED INTO
BITS <N> THRU (N+LENGTH-1) OF <INTO>.

CM REQUIRED: 20B

EXAMPLE

I = 7777 1111 2222 5555 4444B
J = 3333 2222 1111 5555 4436B
AA = SBYT (37, 6, I, J)

RESULTS IN

AA = 7777 1136 2222 5555 4444B
I = 7777 1136 2222 5555 4444B

SUBPROGRAMS REQUIRED
PART OF LANGUAGE
NONE
OTHERS
NONE

AUTHOR: FROM CDC KRONOS SYSTEM

DATE WRITTEN:

DATE(S) REVISED
03/23/83 - RECOMPILE AT NOS/BE LEVEL 552

LOCATION OF DECKS
SOURCE
UPDATE LIBRARY ON MSS: NSRDCPL,UN=CSYS
OBJECT
EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT
BEGIN,DOCGET,,NSRDC,,SBYT,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'SEMICO'

PURPOSE

REPLACE DISPLAY CODE 00B WITH 77B (SEMI-COLON)
03/23/83 - RECOMPILE AT NOS/BE LEVEL 552

FUNCTIONAL CATEGORIES: M4

LANGUAGE: FORTRAN IV EXTENDED

COMPUTERS (OPERATING SYSTEMS)
CDC 6000/CYBER 170 (NOS/BE)

REMARKS
NONE

USAGE
CALL SEMICO (IA, I)

DESCRIPTION OF PARAMETERS
IA - (ARRAY) TO BE PROCESSED
I - NUMBER OF WORDS IN 'IA' TO BE PROCESSED

CM REQUIRED: 21B

SUBPROGRAMS REQUIRED
PART OF LANGUAGE
SHIFT
OTHERS
NONE

AUTHOR
? - NWL

DATE WRITTEN: ?

DATE(S) REVISED
03/23/83 - RECOMPILE AT NOS/BE LEVEL 552

LOCATION OF DECKS
SOURCE
UPDATE LIBRARY ON MSS: NSRDCPL,UN=CSYS
OBJECT
EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT
BEGIN,DOCGET,,NSRDC,,SEMICO,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'SETREW'

PURPOSE

CONVERT REWIND OPTION INTO RM OPEN AND CLOSE CODES

FUNCTIONAL CATEGORIES: M4

LANGUAGE: FORTRAN IV EXTENDED

COMPUTERS (OPERATING SYSTEMS)
CDC 6000/CYBER 170 (NOS/BE)

REMARKS

NONE

USAGE

CALL SETREW (REW, OPEN, CLOSE, NOE)
CALL SETREW (REW, OPEN, CLOSE)

DESCRIPTION OF PARAMETERS

REW - INPUT REWIND OPTION. ONE OF:
A - OPEN=NOREWIND; CLOSE=REWIND
B - OPEN=REWIND ; CLOSE=NOREWIND
E - OPEN=POSITION BEFORE END-OF-INFORMATION;
CLOSE=NOREWIND
EN - OPEN=POSITION BEFORE EOI; CLOSE=NOREWIND
ER - OPEN=POSITION BEFORE EOI; CLOSE=REWIND
EU - OPEN=POSITION BEFORE EOI; CLOSE=UNLOAD
R - OPEN=REWIND ; CLOSE=REWIND
U - OPEN=REWIND ; CLOSE=REWIND AND UNLOAD
OTHER - OPEN=NOREWIND; CLOSE=NOREWIND
(ANY WORDS BEGINNING WITH THESE LETTERS WILL
PRODUCE THE SAME RESULTS. ONLY THE FIRST 1
OR 2 LETTERS ARE RETURNED IN L-FORMAT)
OPEN - WILL CONTAIN OPEN REWIND OPTION (1LE, 1LN, 1LR)
CLOSE - WILL CONTAIN CLOSE REWIND OPTION (1LN, 1LR, 1LU)
NOE - OMITTED OR 0 - ALLOW ALL VALUES OF REW
OTHER - DO NOT ALLOW 'E' VALUES OF REW

CM REQUIRED: 113B

SUBPROGRAMS REQUIRED
PART OF LANGUAGE
LOCF
OTHERS
NONE

AUTHOR
DAVID V SOMMER - DTNSRDC CODE 1892.2

DATE WRITTEN: 10/29/75

DATE(S) REVISED
01/29/76
01/11/76 - ADD 'NOE' PARAMETER
02/15/83 - RECOMPILE AT NOS/BE LEVEL 552

LOCATION OF DECKS
SOURCE
UPDATE LIBRARY ON MSS: NSRDCPL,UN=CSYS
OBJECT
EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT
BEGIN,DOCGET,,NSRDC,,SETREW,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'SHIFTA'

PURPOSE

SHIFT WHOLE ARRAY SPECIFIED NUMBER OF BITS (CROSSING OVER
WORD BOUNDARIES)

FUNCTIONAL CATEGORIES: M4

LANGUAGE: FORTRAN IV EXTENDED

COMPUTERS (OPERATING SYSTEMS)
CDC 6000/CYBER 170 (NOS/BE)

REMARKS

SEE 'AShift' FOR SHIFTING INDIVIDUAL WORDS OF AN ARRAY.

USAGE

CALL SHIFTA (A, B, N, NBITS)

DESCRIPTION OF PARAMETERS

A - INPUT ARRAY OF DIMENSION 'N'
B - OUTPUT ARRAY OF DIMENSION 'N+1'
(MAY NOT BE SAME AS 'A')
N - NUMBER OF WORDS TO BE PROCESSED
NBITS - NUMBER OF BITS TO SHIFT
<0 - SHIFT TO LEFT
(LEFTMOST BITS LOST, TRAILING BITS SET TO 0,
B(N+1) NOT DEFINED)
=0 - JUST MOVE (B(N+1) IS SET TO 0)
>0 - SHIFT TO RIGHT
(LEADING AND TRAILING BITS SET TO 0)

CM REQUIRED: 57B

SUBPROGRAMS REQUIRED

PART OF LANGUAGE
SHIFT
OTHERS
NONE

AUTHOR

DAVID V SOMMER - DTNSRDC CODE 1892.2

DATE WRITTEN: 04/26/74

DATE(S) REVISED

03/23/83 - RECOMPILE AT NOS/BE LEVEL 552

LOCATION OF DECKS

SOURCE
UPDATE LIBRARY ON MSS: NSRDCPL,UN=CSYS
OBJECT
EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC,,SHIFTA,OUTPUT,MSACCES=<PASSWORD>.

FUNCTION 'SIMPUN'

PURPOSE

SIMPSON'S RULE INTEGRATION - EQUAL OR UNEQUAL INTERVALS

FUNCTIONAL CATEGORIES: D1

LANGUAGE: FORTRAN IV

COMPUTERS (OPERATING SYSTEMS)

CDC 6000/CYBER 170 (NOS/BE)

REMARKS

NONE

USAGE

VALUE = SIMPUN (X, Y, N)

DESCRIPTION OF PARAMETERS

X - ARRAY OF MONOTONE X-VALUES

Y - ARRAY OR CORRESPONDING Y-VALUES

N - NUMBER OF VALUES

CM REQUIRED: 100B

ERROR MESSAGE

L=XXXXX, X=X.XXXXXXX E+YY, X NOT MONOTONE STOP
SELF-EXPLANATORY

METHOD

THE INTEGRAL FROM X1 TO XN OF YDX IS EVALUATED BY FITTING
PARABOLAS TO SUCCESSIVE INTERVALS AND INTEGRATING OVER
THE INTERVALS.

SUBPROGRAMS REQUIRED

PART OF LANGUAGE

NONE

OTHERS

NONE

AUTHORS

WERNER FRANK

SHARON E GOOD - DTNSRDC CODE 1892.1

DATE WRITTEN:

DATE(S) REVISED

06/29/68 - SEG

09/20/83 - RECOMPILE AT NOS/BE LEVEL 552

LOCATION OF DECKS

SOURCE

UPDATE LIBRARY ON MSS: NSRDCPM,UN=CSYS (*DECK AMSIUF)

OBJECT

EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC,,SIMPUN,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'SKPFIL'

PURPOSE

REPOSITION A SEQUENTIAL FILE FORWARD OR BACKWARD BY A
SPECIFIED NUMBER OF UNITS (FOR EXISTING RECORDS ONLY)

FUNCTIONAL CATEGORIES: Q3

LANGUAGE: CDC 6000 CP COMPASS

COMPUTERS (OPERATING SYSTEMS)
CDC 6000/CYBER 170 (NOS/BE)

USAGE

CALL SKPFIL (LUN, COUNT)
CALL SKPFIL (LUN, COUNT, TYPE)
CALL SKPFIL (LUN, COUNT, TYPE, RT)
CALL SKPFIL (LUN, COUNT, TYPE, RT, BT)
CALL SKPFIL (LUN, COUNT, TYPE, RT, BT, SB)

DESCRIPTION OF PARAMETERS

LUN - FORTRAN LOGICAL UNIT NUMBER OF FILE
COUNT - NUMBER OF UNITS TO REPOSITION
 <0 - BACKWARD SKIP
 0 - NO SKIP
 >0 - FORWARD SKIP
TYPE - TYPE OF RECORDS TO SKIP
 (1 CHARACTER, LEFT-ADJUSTED)
 B - PARTITIONS
 F - LEVEL 17, EOFs (DEFAULT)
 L - LOGICAL RECORDS
 P - SCOPE LOGICAL RECORDS OF LEVEL 0
 (FOR RT=W, BT=I, A PARTITION BOUNDARY IS
 IMPLEMENTED THRU A RECORD LEVEL ZERO PRU.)
 S - SECTIONS
RT - RECORD TYPE (1 CHARACTER, LEFT-ADJUSTED)
 D - CHARACTER COUNT
 F - FIXED LENGTH
 R - RECORD MARK
 S - SCOPE LOGICAL RECORD (DEFAULT)
 T - TRAILER
 U - USER-DEFINED LENGTH
 W - CONTROL WORD
 Z - ZERO
BT - BLOCK TYPE (1 CHARACTER, LEFT-ADJUSTED)
 C - CHARACTER COUNT (DEFAULT)
 E - EXACT RECORDS
 I - INTERNAL
 K - RECORD COUNT
 (MBL MUST BE PRE-DEFINED)
SB - CIRCULAR BUFFER SUPPRESSION
 =0 - USE CIRCULAR BUFFER (DEFAULT)
 <>0 - USE WORKING STORAGE AREA

NOTE: FORTRAN UNFORMATTED - BT=I, RT=W
FORTRAN FORMATTED - BT=C, RT=Z
BUFFER IN/OUT - BT=C, RT=S

CM REQUIRED: 412B

EXAMPLES

- 1 - SKIP 5 FILES ON 'TAPE10', A STRANGER TAPE:
CALL SKPFIL (10, 5)
- 2 - SKIP A PARTITION ON A FORTRAN UNFORMATTED TAPE:
CALL SKPFIL (10, 1, 1LP, 1LW, 1LI) RECOMMENDED FORM
CALL SKPFIL (10, 1, 1LB, 1LW, 1LI) MORE CP TIME

REMARKS

- 1 - SEQUENTIAL FILES ONLY
- 2 - FOR RT=U: FORWARD SKIP IS NOT ALLOWED
- 3 - FOR TYPE=B/L/S, BACKWARD SKIP IS NOT SUPPORTED FOR:
 - A - RT=D/T
 - B - RT=U AND BT<>K
 - C - BLOCKED FILES OF ONE RECORD PER BLOCK
 - D - BT=E
- 4 - FOR BT=C AND RT=F: IF A BACKWARD SKIP IS ATTEMPTED WHEN A FILE IS POSITIONED AT A TERMINATOR, IT IS NOT POSSIBLE TO DETERMINE THE EXACT RECORD BOUNDARY.
- 5 - FOR RT=F: IF A BACKWARD SKIP IS ATTEMPTED AND THE LENGTH OF EACH RECORD IS NOT A MULTIPLE OF 10, POSITIONING MAY BE UNPREDICTABLE (NOT A RECORD BOUNDARY).

'SKPFIL' TERMINATES IF:

- 1 - THE FILE TO BE SKIPPED HAS NOT BEEN DEFINED
- 2 - THE RECORD MANAGER PROCESSORS NEEDED FOR SKIPPING HAVE NOT BEEN LOADED

SKPFIL IS PART OF THE OPERATING SYSTEM, NOT IN LIBRARY NSRDC. THE DOCUMENT IS HERE FOR CONVENIENCE.

METHOD

- 1 - IF COUNT=0, CONTROL IS RETURNED TO THE USER
- 2 - THE PARAMETERS ARE SCRUTINIZED AND DEFAULT PARAMETERS ARE USED AS REQUIRED
- 3 - CHECK TO BE SURE THAT THE RECORD MANAGER PROCESSORS NEEDED FOR SKIPPING HAVE BEEN LOADED
- 4 - THE RECORD TYPE AND FILE ORGANIZATION ARE SET IN THE FIT
- 5 - IF THE FILE TO BE REPOSITIONED HAS NOT BEEN OPENED, OPEN THE FILE
- 6 - THE COUNT AND TYPE PARAMETERS ARE PLACED IN A RECORD MANAGER SKIPDU MACRO AND THIS MACRO IS EXECUTED, THEREBY REPOSITIONING THE FILE
- 7 - IF A BOUNDARY CONDITION IS DETECTED BEFORE THE SKIP COUNT IS EXHAUSTED, CONTROL IS RETURNED TO THE USER (SEE SUBROUTINE 'SKPSTAT'). THE FILE IS POSITIONED IMMEDIATELY AFTER THE TERMINATOR ON A FORWARD SKIP AND BEFORE THE DELIMITER ON A BACKWARD SKIP.

SUBPROGRAMS REQUIRED

PART OF LANGUAGE

NONE

OTHERS

GETFIT=

AUTHOR

ANTHONY CINCOTTA - DTNSRDC CODE 1892.3

DATE WRITTEN: 03/25/75

DATE(S) REVISED

08/75 01/76

LOCATION OF DECKS

SOURCE

CODE 1892.3

OBJECT

EDITLIB SYSTEM LIBRARY: SYSLIB (NO ATTACH OR SPECIAL
LDSET, LIB= REQUIRED)

ANOTHER COPY OF THIS DOCUMENT

BEGIN, DOCGET, , NSRDC, , SKPFIL, OUTPUT, MSACCES=<PASSWORD>.

SUBROUTINE 'SKPSTAT'

PURPOSE

GET THE STATUS OF THE LAST CALL TO 'SKPFIL'

FUNCTIONAL CATEGORIES: QO

LANGUAGE: CDC 6000 CP COMPASS

COMPUTERS (OPERATING SYSTEMS)

CDC 6000/CYBER 170 (NOS/BE)

USAGE

CALL SKPSTAT (STATUS)

DESCRIPTION OF PARAMETER

STATUS - STATUS OF THE LAST CALL TO 'SKPFIL'

000B - MID-RECORD

001B - END-OF-LABEL-GROUP OR
BEGINNING-OF-INFORMATION (EOL/BOI)

002B - BEGINNING-OF-FILE/VOLUME (BOF/BOV)

004B - END-OF-VOLUME (EOV)

010B - END-OF-SECTION (EOS)

020B - END-OF-RECORD (EOR)

040B - END-OF-PARTITION (EOP)

100B - END-OF-INFORMATION (EOI)

CM REQUIRED: 0B (THIS IS AN ENTRY POINT IN 'SKPFIL',
WHICH REQUIRES 412B)

REMARKS

SKPSTAT IS PART OF THE OPERATING SYSTEM, NOT IN LIBRARY
NSRDC. THE DOCUMENT IS HERE FOR CONVENIENCE.

SUBPROGRAMS REQUIRED

PART OF LANGUAGE

NONE

OTHERS

NONE

AUTHOR

ANTHONY CINCOTTA - DTNSRDC CODE 1892.3

DATE WRITTEN: 03/25/75

DATE(S) REVISED

08/75 01/76

LOCATION OF DECKS

SOURCE

CODE 1892.3

OBJECT

EDITLIB SYSTEM LIBRARY: SYSLIB (NO ATTACH OR SPECIAL
LDSET, LIB= REQUIRED)

ANOTHER COPY OF THIS DOCUMENT

BEGIN, DOCGET, , NSRDC, , SKPSTAT, OUTPUT, MSACCES=<PASSWORD>.

SUBROUTINE 'SKWEZL'

PURPOSE

SQUEEZE LEFT AND REMOVE BLANKS AND OOB

FUNCTIONAL CATEGORIES: M4

LANGUAGE: FORTRAN IV

COMPUTERS (OPERATING SYSTEMS)

CDC 6000/CYBER 170 (NOS/BE)

REMARKS

SEE SKWEZR.

USAGE

CALL SKWEZL (A, NA, NC, NW)

DESCRIPTION OF PARAMETERS

A - ARRAY TO BE SQUEEZED
(WILL BE REPLACED BY SQUEEZED ARRAY)
NA - NUMBER OF WORDS TO BE SQUEEZED
NC - OUTPUT NUMBER OF CHARACTERS IN SQUEEZED ARRAY
NW - OUTPUT NUMBER OF WORDS IN SQUEEZED ARRAY

CM REQUIRED: 73B

SUBPROGRAMS REQUIRED

PART OF LANGUAGE

NONE

OTHERS

GETCHA - EXTRACT CHARACTER FROM ARRAY
PUTCHA - PUT CHARACTER INTO ARRAY

AUTHOR

DAVID V SOMMER - DTNSRDC CODE 1892.2

DATE WRITTEN: 03/19/76

DATE(S) REVISED

03/23/83 - RECOMPILE AT NOS/BE LEVEL 552

LOCATION OF DECKS

SOURCE

UPDATE LIBRARY ON MSS: NSRDCPL,UN=CSYS

OBJECT

EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC,,SKWEZL,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'SKWEZR'

PURPOSE

SQUEEZE RIGHT AND REMOVE BLANKS AND OOB

FUNCTIONAL CATEGORIES: M4

LANGUAGE: FORTRAN IV

COMPUTERS (OPERATING SYSTEMS)

CDC 6000/CYBER 170 (NOS/BE)

REMARKS

SEE SKWEZL.

USAGE

CALL SKWEZR (A, NA, NC, NW)

DESCRIPTION OF PARAMETERS

A - ARRAY TO BE SQUEEZED
(WILL BE REPLACED BY SQUEEZED ARRAY)
NA - NUMBER OF WORDS TO BE SQUEEZED
NC - OUTPUT POSITION OF FIRST NON-ZERO CHARACTERS IN
SQUEEZED ARRAY (POSITION 1 IS LEFTMOST CHARACTER IN
A(1))
NW - OUTPUT SUBSCRIPT OF FIRST NON-ZERO WORD

CM REQUIRED: 75B

SUBPROGRAMS REQUIRED

PART OF LANGUAGE

NONE

OTHERS

GETCHA - EXTRACT CHARACTER FROM ARRAY
PUTCHA - PUT CHARACTER INTO ARRAY

AUTHOR

DAVID V SOMMER - DTNSRDC CODE 1892.2

DATE WRITTEN: 03/19/76

DATE(S) REVISED

03/23/83 - RECOMPILE AT NOS/BE LEVEL 552

LOCATION OF DECKS

SOURCE

UPDATE LIBRARY ON MSS: NSRDCPL,UN=CSYS

OBJECT

EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC,,SKWEZR,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'SNCNDN'

PURPOSE

EVALUATE THE THREE JACOBIAN ELLIPTIC FUNCTIONS

FUNCTIONAL CATEGORIES: C3

LANGUAGE: FORTRAN IV

COMPUTERS (OPERATING SYSTEMS)

CDC 6000/CYBER 170 (NOS/BE)

REMARKS

IF CM=0 AND $ABS(X) > (2K/PI)*6.87E10$, WHERE K IS THE QUARTER PERIOD OF SN, THE ERROR MESSAGE

SNCNDN ARGUMENT X TOO LARGE. X= IS PRINTED ON FILE 'OUTPUT'.

USAGE

CALL SNCNDN (X, CM, SN, CN, DN)

DESCRIPTION OF PARAMETERS

X - INPUT PARAMETER

CM - INPUT PARAMETER

SN - OUTPUT PARAMETER - WILL CONTAIN THE VALUE OF SN(X,K)

CN - OUTPUT PARAMETER - WILL CONTAIN THE VALUE OF CN(X,K)

DN - OUTPUT PARAMETER - WILL CONTAIN THE VALUE OF DN(X,K)

CM REQUIRED: 254B

OUTPUT UNITS

UNIT #	LFN	USE
-----	-----	-----
	OUTPUT	ERROR MESSAGE (SEE REMARKS)

METHOD

GAUSS TRANSFORMATION

REFERENCE

BULIRSCH, R, "NUMERICAL CALCULATIONS OF ELLIPTIC INTEGRALS AND ELLIPTIC FUNCTIONS", NUMERISCHE MATHEMATIK, 7, 1965, PP. 78-90

SUBPROGRAMS REQUIRED

PART OF LANGUAGE

ABS	EXP	SIGN	SIN	SQRT
OTHERS				
NONE				

AUTHOR

R BULIRSCH

DATE WRITTEN: 01/68

DATE(S) REVISED

09/20/83 - RECOMPILE AT NOS/BE LEVEL 552

LOCATION OF DECKS

SOURCE

UPDATE LIBRARY ON MSS: NSRDCPM,UN-CSYS

OBJECT

EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC,,SNCNDN,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'SSORT'

PURPOSE

FTN-CALLABLE SHELL SORT FOR REAL ARRAYS

LANGUAGE: FORTRAN IV EXTENDED

FUNCTIONAL CATEGORIES: M1

COMPUTERS (OPERATING SYSTEMS)

CDC 6000/CYBER 170 (NOS/BE)

REMARKS

THIS MAY BE USED WHEN YOU HAVE DEFINED TWO ARRAYS WHICH
"GO TOGETHER" AND YOU ARE SORTING ONE ARRAY AND WISH TO KEEP
CORRESPONDING ELEMENTS OF THE OTHER ARRAY WITH IT. THAT IS,
WHEREVER A(1) ENDS UP AFTER THE SORT, T(1) WILL BE IN THE
SAME RELATIVE POSITION.

USE ISSORT FOR INTEGERS.

USAGE

CALL SSORT (A, I, T)

CALL SSORT (A, I)

DESCRIPTION OF PARAMETERS

A - REAL ARRAY TO BE SORTED

I - NUMBER OF ELEMENTS TO BE SORTED

T - IF PRESENT, AN ASSOCIATED ARRAY RE-ORDERED TO MAINTAIN
1 TO 1 CORRESPONDENCE WITH THE ELEMENTS OF ARRAY 'A'

CM REQUIRED: 70B

SUBPROGRAMS REQUIRED

PART OF LANGUAGE

LOCF SHIFT

OTHERS

NONE

AUTHOR

C FLINK - KPS NWL

DATE WRITTEN: 12/07/70

DATE(S) REVISED

03/23/83 - RECOMPILE AT NOS/BE LEVEL 552

LOCATION OF DECKS

SOURCE

UPDATE LIBRARY ON MSS: NSRDCPL,UN=CSYS

OBJECT

EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC,,SSORT,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'SSORTF'

PURPOSE

FTN-CALLABLE SHELL SORT FOR TWO-DIMENSIONAL REAL ARRAYS

FUNCTIONAL CATEGORIES: M1

LANGUAGE: FORTRAN IV

COMPUTERS (OPERATING SYSTEMS)

CDC 6000/CYBER 170 (NOS/BE)

REMARKS

THIS ROUTINE IS INEFFICIENT IF M .GT. 10.

USAGE

CALL SSORTF (A, TEMP, M, N, 1)

CALL SSORTF (A, TEMP, M, N)

DESCRIPTION OF PARAMETERS

A - REAL ARRAY TO BE SORTED

TEMP - TEMPORARY ARRAY OF DIMENSION M USED IN THE SORT

M - NUMBER OF WORDS PER ITEM

N - NUMBER OF ITEMS PER ARRAY

(DIMENSION OF A IS A(M,N))

I - IF PRESENT, NUMBER FROM 1 TO M SPECIFYING ON WHICH
WORD OF AN ITEM TO SORT.

IF OMITTED, I=1.

CM REQUIRED: 200B

SUBPROGRAMS REQUIRED

PART OF LANGUAGE

LOCF SHIFT

OTHERS

MOVECM - MOVE AN ARRAY

AUTHOR

C FLINK - KPS NWL

DATE WRITTEN: 01/10/71

DATE(S) REVISED

11/23/76 - DVS - DTNSRDC - CHANGE SUBROUTINE SENT TO MOVLEV

02/21/80 - DVS - DTNSRDC - CHANGE MOVLEV TO MOVECM

03/23/83 - RECOMPILE AT NOS/BE LEVEL 552

LOCATION OF DECKS

SOURCE

UPDATE LIBRARY ON MSS: NSRDCPL,UN=CSYS

OBJECT

EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC,,SSORTF,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'SSORTI'

PURPOSE

FTN-CALLABLE SHELL SORT FOR TWO-DIMENSIONAL INTEGER ARRAYS

FUNCTIONAL CATEGORIES: M1

LANGUAGE: FORTRAN IV EXTENDED

COMPUTERS (OPERATING SYSTEMS)

CDC 6000/CYBER 170 (NOS/BE)

REMARKS

THIS ROUTINE IS INEFFICIENT IF M .GT. 10.

USAGE

CALL SSORTI (A, TEMP, M, N, I)

CALL SSORTI (A, TEMP, M, N)

DESCRIPTION OF PARAMETERS

A - INTEGER ARRAY TO BE SORTED

TEMP - TEMPORARY ARRAY OF DIMENSION M USED IN THE SORT

M - NUMBER OF WORDS PER ITEM

N - NUMBER OF ITEMS PER ARRAY

(DIMENSION OF A IS A(M,N))

I - IF PRESENT, NUMBER FROM 1 TO M SPECIFYING ON WHICH
WORD OF AN ITEM THE ARRAY IS TO BE SORTED.

IF ABSENT, THE ARRAY WILL BE SORTED ON THE FIRST
WORD (I=1).

CM REQUIRED: 200B

SUBPROGRAMS REQUIRED

PART OF LANGUAGE

LOCF SHIFT

OTHERS

MOVECM - MOVE AN ARRAY

AUTHOR

C FLINK - KPS NWL

ALBAN P GASS - NWL

DAVID V SOMMER - DTNSRDC CODE 1892.2

DATE WRITTEN: 01/10/71

DATE(S) REVISED

03/10/74 - APG - CHANGE FROM REAL TO INTEGER

06/09/76 - DVS - CHANGE SUBROUTINE SENT TO MOVLEV

02/21/80 - DVS - CHANGE SUBROUTINE MOVLEV TO MOVECM

03/23/83 - RECOMPILE AT NOS/BE LEVEL 552

LOCATION OF DECKS

SOURCE

UPDATE LIBRARY ON MSS: NSRDCPL,UN=CSYS

OBJECT

EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC,,SSORTI,OUTPUT,MSACCES=<PASSWORD>.

02/10/84

2-279

SSORTI - 1 OF 1

SUBROUTINE 'SSORTL'

PURPOSE

FTN-CALLABLE LOGICAL SHELL SORT FOR CHARACTER ARRAYS

FUNCTIONAL CATEGORIES: M1

LANGUAGE: FORTRAN IV EXTENDED

COMPUTERS (OPERATING SYSTEMS)

CDC 6000/CYBER 170 (NOS/BE)

REMARKS

THIS MAY BE USED WHEN YOU HAVE DEFINED TWO ARRAYS WHICH
"GO TOGETHER" AND YOU ARE SORTING ONE ARRAY AND WISH TO KEEP
CORRESPONDING ELEMENTS OF THE OTHER ARRAY WITH IT. THAT IS,
WHEREVER A(I) ENDS UP AFTER THE SORT, T(I) WILL BE IN THE
SAME RELATIVE POSITION.

USAGE

CALL SSORTL (A, I, M, T)

CALL SSORTL (A, I, M)

DESCRIPTION OF PARAMETERS

A - CHARACTER ARRAY TO BE SORTED

I - NUMBER OF ELEMENTS IN ARRAY 'A' TO BE SORTED

M - MASK WORD WITH THE RELEVANT BITS SET

T - IF PRESENT, ASSOCIATED ARRAY, RE-ORDERED SUCH THAT

A(K) STILL RELATES TO T(K)

CM REQUIRED: 133B

SUBPROGRAMS REQUIRED

PART OF LANGUAGE

LOCF

SHIFT

OTHERS

EQU60

AUTHOR

C FLINK - KPS NWL

DATE WRITTEN: 12/03/70

DATE(S) REVISED

03/23/83 - RECOMPILE AT NOS/BE LEVEL 552

LOCATION OF DECKS

SOURCE

UPDATE LIBRARY ON MSS: NSRDCPL,UN=CSYS

OBJECT

EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC,,SSORTL,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'SSORT3'

PURPOSE

FTN-CALLABLE SHELL SORT FOR REAL ARRAYS WITH ASSOCIATED REAL
ARRAY AND INTEGER ARRAY

LANGUAGE: FORTRAN IV EXTENDED

FUNCTIONAL CATEGORIES: M1

COMPUTERS (OPERATING SYSTEMS)
CDC 6000/CYBER 170 (NOS/BE)

REMARKS

SEE SUBROUTINE 'SSORT'.

USAGE

CALL SSORT3 (ARRAY, NWORDS, REALA, INTA)
CALL SSORT3 (ARRAY, NWORDS)

DESCRIPTION OF PARAMETERS

ARRAY - REAL ARRAY TO BE SORTED
NWORDS - NUMBER OF ELEMENTS TO BE SORTED
REALA - IF PRESENT, AN ASSOCIATED REAL ARRAY RE-ORDERED TO
MAINTAIN 1-TO-1 CORRESPONDENCE WITH THE ELEMENTS
OF ARRAY 'ARRAY'
INTA - AN ASSOCIATED INTEGER ARRAY RE-ORDERED TO MAINTAIN
1-TO-1 CORRESPONDENCE WITH THE ELEMENTS OF ARRAY
'ARRAY'. IF 'REALA' IS PRESENT, 'INTA' MUST ALSO
BE PRESENT. IF 'REALA' IS OMITTED, 'INTA' MUST
ALSO BE OMITTED.

CM REQUIRED: 111B

SUBPROGRAMS REQUIRED

PART OF LANGUAGE
LOCF SHIFT
OTHERS
NONE

AUTHORS

C FLINK - KPS NWL
D V SOMMER - DTNSRDC (ADD INTEGER ARRAY)

DATE WRITTEN: 12/07/70

DATE(S) REVISED

02/02/82 - ADD INTEGER ARRAY
03/23/83 - RECOMPILE AT NOS/BE LEVEL 552

LOCATION OF DECKS

SOURCE
UPDATE LIBRARY ON MSS: NSRDCPL, UN=CSYS
OBJECT
EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT

BEGIN, DOCGET, , NSRDC, , SSORT3, OUTPUT, MSACCES=<PASSWORD>.

FUNCTION 'SUMIT'

PURPOSE

SUM ELEMENTS OF REAL ARRAY

FUNCTIONAL CATEGORIES: A1

LANGUAGE: FORTRAN IV

COMPUTERS (OPERATING SYSTEMS)

CDC 6000/CYBER 170 (NOS/BE)

REMARKS

NONE

USAGE

ITOTAL = SUMIT (ARRAY, N)

DESCRIPTION OF PARAMETERS

SUMIT - WILL CONTAIN $ARRAY(1)+ARRAY(2)+\dots+ARRAY(N)$

ARRAY - ARRAY TO BE SUMMED

N - NUMBER OF ELEMENTS OF ARRAY TO BE SUMMED

CM REQUIRED: 16B

SUBPROGRAMS REQUIRED

PART OF LANGUAGE

NONE

OTHERS

NONE

AUTHOR

DAVID V SOMMER - DTNSRDC CODE 1892.2

DATE WRITTEN: 11/23/76

DATE(S) REVISED

03/23/83 - RECOMPILE AT NOS/BE LEVEL 552

LOCATION OF DECKS

SOURCE

UPDATE LIBRARY ON MSS: NSRDCPL,UN=CSYS

OBJECT

EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC,,SUMIT,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'SWAP'

PURPOSE

SWAP TWO ARRAYS

FUNCTIONAL CATEGORIES: K2

LANGUAGE: FORTRAN IV EXTENDED

COMPUTERS (OPERATING SYSTEMS)

CDC 6000/CYBER 170 (NOS/BE)

REMARKS

NONE

USAGE

CALL SWAP (A, B, NWORDS)

DESCRIPTION OF PARAMETERS

A,B - ARRAYS TO BE SWAPPED

NWORDS - NUMBER OF WORDS TO BE SWAPPED

CM REQUIRED: 16B

EXAMPLE

```
PROGRAM TEST (OUTPUT=128)
INTEGER A(10), B(10)
DATA A/ 1, 2, 3, 4, 5, 6, 7, 8, 9, 10/
DATA B/ 10, 9, 8, 7, 6, 5, 4, 3, 2, 1/
...
CALL SWAP (A, B, 10)
C   ARRAY A NOW CONTAINS 10, 9, 8, 7, 6, 5, 4, 3, 2, 1
C   ARRAY B NOW CONTAINS 1, 2, 3, 4, 5, 6, 7, 8, 9, 10
...
```

SUBPROGRAMS REQUIRED

PART OF LANGUAGE

NONE

OTHERS

NONE

AUTHOR

DAVID V SOMMER - DTNSRDC CODE 1892.2

DATE WRITTEN: 11/12/80

DATE(S) REVISED

03/23/83 - RECOMPILE AT NOS/BE LEVEL 552

LOCATION OF DECKS

SOURCE

UPDATE LIBRARY ON MSS: NSRDCPL,UN=CSYS

OBJECT

EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC,,SWAP,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'TEKTRI'

PURPOSE

INITIALIZE COMMON BLOCK /TEKTRN/ WITH ASCII CONTROL CODES
FOR THE TEKTRONIX GRAPHICS TERMINALS

FUNCTIONAL CATEGORIES: M4

LANGUAGE: FORTRAN 77 EXTENDED

COMPUTERS (OPERATING SYSTEMS)

CDC 6000/CYBER 170 (NOS/BE)

REMARKS

THIS SUBROUTINE MUST BE EXECUTED PRIOR TO GENERATING ASCII
MESSAGES WITH TEKTRONIX CONTROL CODES USING SUBROUTINE
ASCII.

COMMON BLOCK /TETRN/ IS OBTAINED BY RUNNING PROCEDURE ASCIIIO
AND INSERTING THE COMMON BLOCK INTO EACH (SUB)PROGRAM WHICH
WILL GENERATE ASCII MESSAGES HAVING TEKTRONIX CONTROL CODES.

USAGE

CALL TEKTRI

CM REQUIRED: 6B

NAMES OF TEKTRONIX CONTROL CODES

CHARACTER SETTING

SELASC - SELECT ASCII CHARACTERS
SELAPL - SELECT APL CHARACTERS
C74 - SELECT 74 CHARACTERS PER LINE, 35 LINES
C81 - SELECT 81 CHARACTERS PER LINE, 38 LINES
C121 - SELECT 121 CHARACTERS PER LINE, 58 LINES
C133 - SELECT 133 CHARACTERS PER LINE, 64 LINES

MODE SETTING

SETALF - SET ALPHA MODE (NORMAL)
SETGIN - SET GIN MODE
SETGRA - SET GRAPH MODE
SETLCE - SET LCE
BYPASS - SET BYPASS
INCPLT - SET INCREMENTAL PLOT

MISCELLANEOUS

CLRSCR - HOME, CLEAR SCREEN
HRDCPY - MAKE HARD COPY
REVLf - REVERSE LINE FEED

BEAM AND VECTOR SELECTORS

NZNVA - NORMAL Z AXIS AND NORMAL VECTORS OR ALPHA
NZDLIN - NORMAL Z AXIS AND DOTTED LINE VECTORS
NZDDV - NORMAL Z AXIS AND DOT-DASHED VECTORS
NZSDV - NORMAL Z AXIS AND SHORT-DASHED VECTORS
NZLDV - NORMAL Z AXIS AND LONG-DASHED VECTORS
NORMZ - NORMAL Z AXIS
DZNVA - DEFOCUSED Z AXIS AND NORMAL VECTORS OR ALPHA
DZDLIN - DEFOCUSED Z AXIS AND DOTTED LINE VECTORS
DZDDV - DEFOCUSED Z AXIS AND DOT-DASHED VECTORS
DZSDV - DEFOCUSED Z AXIS AND SHORT-DASHED VECTORS
DZLDV - DEFOCUSED Z AXIS AND LONG-DASHED VECTORS
DEFOCZ - DEFOCUSED Z AXIS
WTNVA - WRITE-THRU MODE AND NORMAL VECTORS OR ALPHA
WTDLIN - WRITE-THRU MODE AND DOTTED LINE VECTORS
WTDDV - WRITE-THRU MODE AND DOT-DASHED VECTORS
WTSdv - WRITE-THRU MODE AND SHORT-DASHED VECTORS
WTLDV - WRITE-THRU MODE AND LONG-DASHED VECTORS
WRITRU - WRITE-THRU MODE

INCREMENTAL PLOT MODE CHARACTERS

BEAMOF - BEAM OFF (PEN UP)
BEAMON - BEAM ON (PEN DOWN)
NORTH - MOVE NORTH
NOREST - MOVE NORTHEAST
EAST - MOVE EAST
SOUEST - MOVE SOUTHEAST
SOUTH - MOVE SOUTH
SOUWST - MOVE SOUTHWEST
WEST - MOVE WEST
NORWST - MOVE NORTHWEST

SUBPROGRAMS REQUIRED

PART OF LANGUAGE
NONE
OTHERS
NONE

AUTHOR

DAVID V SOMMER - DTNSRDC CODE 1892.2

DATE WRITTEN: 03/07/84

DATE(S) REVISED

LOCATION OF DECKS

SOURCE DECK
UPDATE LIBRARY ON MSS: NSRDCPL,UN=CSYS
OBJECT DECK
EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC,,TEKTRI,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'TIMLEFT'

PURPOSE

DETERMINE CP (AND IO) TIME LEFT SINCE START OF BATCH JOB
OR INTERCOM COMMAND

FUNCTIONAL CATEGORIES: Q0

LANGUAGE: FORTRAN IV EXTENDED

COMPUTERS (OPERATING SYSTEMS)
CDC 6000/CYBER 170 (NOS/BE)

REMARKS
NONE

USAGE

CALL TIMLEFT (CP, XIO)
CALL TIMLEFT (CP)

DESCRIPTION OF PARAMETERS

CP - WILL CONTAIN CP TIME REMAINING
XIO - IF PRESENT, WILL CONTAIN IO TIME REMAINING
(IF NEGATIVE, THE SYSTEM IS NOT TESTING IO TIME.
THE TOTAL IO TIME USED IS ABS(XIO).)

CM REQUIRED: 65B

SUBPROGRAMS REQUIRED

PART OF LANGUAGE
AND FLOAT SHIFT
OTHERS
RCPA - READ CONTROL POINT AREA

ARITHMETIC STATEMENT FUNCTIONS

R65FMT - FAST R-FORMAT DECODE (RIGHT-ADJ, ZERO-FILLED)

AUTHOR

DAVID V SOMMER - DTNSRDC CODE 1892.2

DATE WRITTEN: 10/27/77

DATE(S) REVISED

03/23/83 - RECOMPILE AT NOS/BE LEVEL 552

LOCATION OF DECKS

SOURCE
UPDATE LIBRARY ON MSS: NSRDCPL,UN=CSYS
OBJECT
EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC,,TIMLEFT,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'TRAILBZ'

PURPOSE

CHANGE TRAILING BLANKS TO ZEROS (OOB)

FUNCTIONAL CATEGORIES: M4

LANGUAGE: FORTRAN IV EXTENDED

COMPUTERS (OPERATING SYSTEMS)

CDC 6000/CYBER 170 (NOS/BE)

REMARKS

OOB IS TREATED AS A BLANK.

THIS SUBROUTINE IS USEFUL WHEN GENERATING MESSAGES FOR PRINTING IN THE DAYFILE USING 'CALL REMARK'. AFTER A MESSAGE IS GENERATED WITH AN ENCODE, A CALL TO 'TRAILBZ' WILL REMOVE ANY TRAILING BLANKS. THIS WILL RESULT IN THE SHORTEST POSSIBLE MESSAGE. THIS IS PARTICULARLY DESIRABLE FOR PROGRAMS WHICH ARE RUN FROM TELETYPE, SINCE TRAILING BLANKS ARE NOT SUPPRESSED FOR DAYFILE MESSAGES.

USAGE

CALL TRAILBZ (A, N)

CALL TRAILBZ (A, N, NW)

CALL TRAILBZ (A, N, NW, NC)

DESCRIPTION OF PARAMETERS

A - ARRAY TO BE PROCESSED

N - NUMBER OF WORDS OF 'A' TO BE PROCESSED

NW - NUMBER OF LAST NON-BLANK WORD OF 'A'

(0 LE NW LE N)

(NW=0 MEANS ALL OF 'A' IS BLANK)

NC - POSITION OF LAST NON-BLANK CHARACTER OF A(NW)

(0 LE NC LE 10)

(NC=0 MEANS ALL OF 'A' IS BLANK)

CM REQUIRED: 72B

SUBPROGRAMS REQUIRED
PART OF LANGUAGE
LOCF MASK SHIFT
OTHERS
NONE

AUTHOR
DAVID V SOMMER - DTNSRDC CODE 1892.2

DATE WRITTEN: 04/08/75

DATE(S) REVISED
12/20/82 - RECOMPILE AT 552 TO REDUCE MEMORY REQUIREMENT

LOCATION OF DECKS
SOURCE
UPDATE LIBRARY ON MSS: NSRDCPL,UN=CSYS
OBJECT
EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT
BEGIN,DOCGET,,NSRDC,,TRAILBZ,OUTPUT,MSACCES=<PASSWORD>.

FUNCTION 'UNHEX3'

PURPOSE

SPREAD 2 CHARACTERS INTO 3 HEX DIGITS

FUNCTIONAL CATEGORIES: M2

LANGUAGE: FORTRAN IV EXTENDED

COMPUTERS (OPERATING SYSTEMS)

CDC 6000/CYBER 170 (NOS/BE)

REMARKS

'UNHEX3' IS AN INTEGER FUNCTION.

WRITTEN TO CHANGE 2-CHARACTER INTERNAL TERMINAL ID INTO
3-CHARACTER (HEX) TERMINAL ID

USAGE

UNHEX3 (INTTID)

DESCRIPTION OF PARAMETERS

INTTID - INPUT INTERNAL TID (E.G., 2L@D)

UNHEX3 - OUTPUT IN FIRST 3 CHARACTERS (E.G., 3LF04)

CM REQUIRED: 43B

SUBPROGRAMS REQUIRED

PART OF LANGUAGE

AND OR SHIFT

OTHERS

NONE

AUTHOR

DAVID V SOMMER - DTNSRDC CODE 1892.2

DATE WRITTEN: 09/19/78

DATE(S) REVISED

03/21/83 - RECOMPILE AT NOS/BE LEVEL 552

LOCATION OF DECKS

SOURCE

UPDATE LIBRARY ON MSS: NSRDCPL,UN=CSYS

OBJECT

EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC,,UNHEX3,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'UNLOAD'

PURPOSE

UNLOAD A FORTRAN FILE

FUNCTIONAL CATEGORIES: Q3

LANGUAGE: FORTRAN IV EXTENDED

COMPUTERS (OPERATING SYSTEMS)

CDC 6000/CYBER 170 (NOS/BE)

REMARKS

THE FILE TO BE UNLOADED MUST BE LISTED IN THE FORTRAN IV PROGRAM STATEMENT.

FORTRAN SEQUENTIAL FILES SHOULD HAVE THEIR BUFFERS FLUSHED BY ISSUING A REWIND BEFORE CALLING THIS ROUTINE.

LEVEL 508 ON: PROGRAMS WHICH REWIND A FILE AND CALL UNLOAD BUT WHICH NEVER READ OR WRITE ON THE FILE MUST BE MODIFIED TO READ (OR WRITE) AT LEAST ONE RECORD BEFORE CALLING UNLOAD. THIS IS BECAUSE THE SYSTEM NO LONGER ALLOWS A CLOSE TO BE ISSUED TO AN UNOPENED (OR NON-EXISTENT) FILE. A FORTRAN REWIND STATEMENT ONLY SETS A FLAG FOR THE NEXT READ/WRITE TO THE FILE. IT DOES NOT CAUSE AN IMMEDIATE REWIND.

DO NOT USE WITH FORTRAN 77; USE THE 'CLOSE' STATEMENT INSTEAD.

USAGE

CALL UNLOAD (IUNIT)

DESCRIPTION OF PARAMETER

IUNIT - FORTRAN LOGICAL UNIT NUMBER

-OR-

1- TO 7-CHARACTER LOCAL FILE NAME

CM REQUIRED: 70B

SUBPROGRAMS REQUIRED

PART OF LANGUAGE

AND CLOSEM LOCF OR

OTHERS

FITADDR - GET FORTRAN FIT ADDRESS

ISTAPE - FORM 'TAPEN' FROM N

LASTC - FIND LAST NON-BLANK CHARACTER

LJBBAM - LDSET, LIB=BAMLIB

TRAILBZ - CHANGE TRAILING BLANKS TO 00B

ARITHMETIC STATEMENT FUNCTIONS

LETTER - TEST FOR LETTER

OCFLAG - GET OPEN/CLOSE FLAG FROM FIT

OPEN - TEST FOR FILE OPEN

R11FMT - FAST R-FORMAT DECODE (RIGHT-ADJ, ZERO-FILLED)

AUTHOR

DAVID V SOMMER - DTNSRDC CODE 1892.2

DATE WRITTEN: 03/07/75

DATE(S) REVISED

07/07/81 - TOTAL REWRITE FOR LEVEL 508

07/14/81 - ADD DUMMY CALL TO SET LDSET,LIB=BAMLIB

07/17/81 - FIX PROBLEM WITH OCCASIONAL REDEFINITION OF
SUBROUTINE ARGUMENT

02/15/83 - RECOMPILE AT NOS/BE LEVEL 552.

LOCATION OF DECKS

SOURCE

UPDATE LIBRARY ON MSS: NSRDCPL,UN=CSYS

OBJECT

EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC,,UNLOAD,OUTPUT,MSACCES=<PASSWORD>.

FUNCTION 'VALDAT'

PURPOSE

LOGICAL FUNCTION TO VALIDATE A DATE FORMAT

FUNCTIONAL CATEGORIES: M4

LANGUAGE: FORTRAN IV EXTENDED

COMPUTERS (OPERATING SYSTEMS)

CDC 6000/CYBER 170 (NOS/BE)

REMARKS

'VALDAT' MUST BE DECLARED LOGICAL IN THE CALLING PROGRAM.

UPON RETURN, IF THE FORMAT WAS VALID, THE DATE IS RETURNED
AS ' MM/DD/YY '.

USAGE

VALDAT (DATE)

DESCRIPTION OF PARAMETERS.

DATE - DATE TO BE ANALYZED
(IF FORMAT OK, RETURNED AS ' MM/DD/YY ')

VALDAT - WILL CONTAIN
.TRUE. - DATE FORMAT WAS OK
.FALSE. - DATE FORMAT WAS NOT OK

CM REQUIRED: 162B

METHOD

DATE FORMAT IS VALIDATED BY THE FOLLOWING CHECKS:
EXACTLY 2 SLASHES
SLASHES SEPARATED BY 1 OR 2 CHARACTERS
SLASHES NOT IN POSITIONS 1, 9 OR 10
MONTH CONTAINS 1 OR 2 DIGITS (LEADING BLANKS OK)
DAY CONTAINS 1 OR 2 DIGITS (LEADING BLANKS OK)
YEAR CONTAINS 2 DIGITS
VALDAT RETURNS IF ANY CHECK FAILS.

SUBPROGRAMS REQUIRED

PART OF LANGUAGE

AND

OR

SHIFT

OTHERS

NONE

AUTHOR

DAVID V SOMMER - DTNSRDC CODE 1892.2

DATE WRITTEN: 07/26/77

DATE(S) REVISED

LOCATION OF DECKS

SOURCE

UPDATE LIBRARY ON MSS: NSRDCPL,UN=CSYS

OBJECT

EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC,,VALDAT,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'VALIDT'

PURPOSE

VALIDATE ARRAY 'A' TO SEE THAT EACH ELEMENT IS ONE OF THOSE
OF ARRAY 'V'

FUNCTIONAL CATEGORIES: M5

LANGUAGE: FORTRAN IV EXTENDED

COMPUTERS (OPERATING SYSTEMS)
CDC 6000/CYBER 170 (NOS/BE)

REMARKS
NONE

USAGE
CALL VALIDT (A, NA, V, NV, VALID)

DESCRIPTION OF PARAMETERS

A - ARRAY TO BE VALIDATED
NA - NUMBER OF ELEMENTS OF 'A' TO BE TESTED
V - ARRAY OF VALID ELEMENTS
NV - NUMBER OF ELEMENTS IN 'V'
VALID - LOGICAL OUTPUT CODE
TRUE - ALL ELEMENTS OF 'A' ARE VALID
FALSE - AT LEAST 1 ELEMENT OF 'A' IS INVALID

CM REQUIRED: 25B

SUBPROGRAMS REQUIRED
PART OF LANGUAGE
NONE
OTHERS
NONE

AUTHOR
DAVID V SOMMER - DTNSRDC CODE 1892.2

DATE WRITTEN: 10/72

DATE(S) REVISED
03/23/83 - RECOMPILE AT NOS/BE LEVEL 552

LOCATION OF DECKS
SOURCE
UPDATE LIBRARY ON MSS: NSRDCPL,UN=CSYS
OBJECT
EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT
BEGIN,DOCGET,,NSRDC,,VALIDT,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'VFILL'

PURPOSE

FILL AN ARRAY WITH USER-SPECIFIED WORD

FUNCTIONAL CATEGORIES: M4

LANGUAGE: CDC 6000 CP COMPASS

COMPUTERS (OPERATING SYSTEMS)

CDC 6000/CYBER 170 (NOS/BE)

REMARKS

NONE

USAGE

CALL VFILL (WORD, A, NA)

DESCRIPTION OF PARAMETERS

WORD - WORD TO BE PUT INTO ARRAY 'A'

A - ARRAY TO RECEIVE 'WORD'

NA - NUMBER OF WORDS IN 'A' TO BE SET TO 'WORD'

CM REQUIRED: 15B

SUBPROGRAMS REQUIRED

PART OF LANGUAGE

NONE

OTHERS

NONE

AUTHOR

C FLINK - KPS NWL

DATE WRITTEN: 02/10/71

DATE(S) REVISED

??/??/74 - DAVID V SOMMER - DTNSRDC CODE 1892.2
(NAME CHANGED FROM 'MOVE' TO 'VFILL')

05/01/79 - MOVE TO BURROUGHS B7700
(CHANGE TO FORTRAN - DVS)

03/23/83 - RECOMPILE AT NOS/BE LEVEL 552

LOCATION OF DECKS

SOURCE

UPDATE LIBRARY ON MSS: NSRDCPL, UN=CSYS

OBJECT

EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC,,VFILL,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'VT1001'

PURPOSE

INITIALIZE COMMON BLOCK /VT100/ WITH ASCII CONTROL CODES
FOR THE DEC VT100 CRT

FUNCTIONAL CATEGORIES: M4

LANGUAGE: FORTRAN 77 EXTENDED

COMPUTERS (OPERATING SYSTEMS)
CDC 6000/CYBER 170 (NOS/BE)

REMARKS

THIS SUBROUTINE MUST BE EXECUTED PRIOR TO GENERATING ASCII
MESSAGES WITH VT100 CONTROL CODES USING SUBROUTINE ASCII.

COMMON BLOCK /VT100/ IS OBTAINED BY RUNNING PROCEDURE ASCII0
AND INSERTING THE COMMON BLOCK INTO EACH (SUB)PROGRAM WHICH
WILL GENERATE ASCII MESSAGES HAVING VT100 CONTROL CODES.

USAGE

CALL VT1001

CM REQUIRED: 6B

NAMES OF VT100 CONTROL CODES

CHARACTER VIDEO ATTRIBUTES

SGR0 - PRIMARY (NORMAL)
SGR1 - BOLD
SGR4 - UNDERSCORE
SGR5 - BLINK
SGR7 - REVERSE

CHARACTER SETS

SCSUKO - UK (GO SET)
SCSASO - ASCII (GO SET)
SCSSGO - SPECIAL GRAPHICS (GO SET)
SCSACO - ALT ROM (GO SET)
SCSAGO - ALT ROM SPECIAL GRAPHICS (GO SET)
SCSUKI - UK (GI SET)
SCSASI - ASCII (GI SET)
SCSSGI - SPECIAL GRAPHICS (GI SET)
SCSACI - ALT ROM (GI SET)
SCSAGI - ALT ROM SPECIAL GRAPHICS (GI SET)

CHARACTER SIZE

PCDHLU - DOUBLE HIGH UPPER HALF
PCDHLL - DOUBLE HIGH BOTTOM HALF
PCSWL - SINGLE WIDTH SINGLE HEIGHT
PCDWL - DOUBLE WIDTH SINGLE HEIGHT

CURSOR MOVEMENTS

CUU - CURSOR UP
(YOU MAY SET THE NUMBER OF ROWS TO MOVE UP
01-24 IN POSITIONS 3-4 - USE SUBROUTINE
ASCPUT)

CUD - CURSOR DOWN
(YOU MAY SET THE NUMBER OF ROWS TO MOVE DOWN
01-24 IN POSITIONS 3-4 - USE SUBROUTINE
ASCPUT)

CUF - CURSOR RIGHT (FORWARD)
(YOU MAY SET THE NUMBER OF ROWS TO MOVE RIGHT
001-132 IN POSITIONS 3-5 - USE SUBROUTINE
ASCPUT)

CUB - CURSOR LEFT (BACK)
(YOU MAY SET THE NUMBER OF ROWS TO MOVE LEFT
001-132 IN POSITIONS 3-5 - USE SUBROUTINE
ASCPUT)

HOME - CURSOR TO HOME POSITION

CUP - POSITION CURSOR RELATIVELY AT I,J
(YOU MUST SET I,J - I IS 01-24 AND IS IN
POSITIONS 3-4; J IS 001-132 AND IS IN
POSITIONS 6-8 - USE SUBROUTINE ASCPUT)

HVP - POSITION CURSOR ABSOLUTELY AT I,J
(YOU MUST SET I,J - I IS 01-24 AND IS IN
POSITIONS 3-4 J IS 001-132 AND IS IN
POSITIONS 6-8 - USE SUBROUTINE ASCPUT)

IND - INDEX (DOWN ONE LINE, SAME COLUMN)

RI - REVERSE INDEX (UP ONE LINE, SAME COLUMN)

NEL - NEW LINE

PCSC - SAVE CURSOR AND ATTRIBUTES (PRIVATE CODE)

PCRC - RESTORE CURSOR AND ATTRIBUTES (PRIVATE CODE)

A NOTE ON SETTING THE CURSOR MOTION COUNTS (CUU,
CUD, CUF, CUB, CUP, HVP): THESE ARE PRE-DEFINED
WITH VALUES OF 01 OR 001 CURSOR POSITION. TO
CHANGE, FOR EXAMPLE, CUD TO 12 ROWS DOWN, USE
CALL ASCII (NCOL, BUF, CUD)
CALL ASCPUT (NCOL-3, BUF, TWO)
CALL ASCPUT (NCOL-2, BUF, ONE)

TO CHANGE CUP TO 05,123, USE
CALL ASCII (NCOL, BUF, CUP)
CALL ASCPUT (NCOL-7, BUF, ZERO)
CALL ASCPUT (NCOL-6, BUF, FIVE)
CALL ASCPUT (NCOL-4, BUF, ONE)
CALL ASCPUT (NCOL-3, BUF, TWO)
CALL ASCPUT (NCOL-2, BUF, THREE)

ERASE

ELO - CURSOR TO END-OF-LINE

EL1 - BEGINNING-OF-LINE TO CURSOR

EL2 - ENTIRE LINE

ED0 - CURSOR TO END-OF-SCREEN

ED1 - BEGINNING-OF-SCREEN TO CURSOR

ED2 - ENTIRE SCREEN

LED'S

PCLL0 - EXTINGUISH ALL
PCLL1 - L1 ON
PCLL2 - L2 ON
PCLL3 - L3 ON
PCLL4 - L4 ON

MEDIA COPY

MC0 - COPY CONTENTS OF DISPLAY TO AUX. OUTPUT DEVICE
MC4 - START COPYING DATA STREAM TO AUX. OUTPUT DEVICE
MC5 - STOP COPYING DATA STREAM TO AUX. OUTPUT DEVICE

MODES

PCARMS - SET AUTO-REPEAT
PCARMR - RESET AUTO-REPEAT
PCCOLS - SET 132 CHARACTER PER LINE DISPLAY
PCCOLR - SET 80 CHARACTER PER LINE DISPLAY
PCKKMS - FOUR CURSOR FUNCTION KEYS SEND APPLICATION
FUNCTIONS
PCKKMR - FOUR CURSOR FUNCTION KEYS SEND ANSI CURSOR
CONTROL COMMANDS
PCCNLS - INTERLACE ON (480 LINES PER FRAME)
(HIGH RESOLUTION GRAPHICS)
PCCNLR - INTERLACE OFF (240 LINES PER FRAME)
(NORMAL DATA DISPLAY)
PCKPAM - KEY PAD APPLICATION MODE
PCKPNM - KEY PAD NUMERIC MODE
LNMS - LF CAUSES CURSOR TO MOVE TO FIRST POSITION OF
NEXT LINE AND CR SEND CR/LF
LNMR - LF CAUSES VERTICAL MOVEMENT AND CR SENDS ONLY
CARRIAGE RETURN CONTROL CODE
PCOMS - ORIGIN MODE (CURSOR POSITION NUMBERS ARE
RELATIVE TO TOP MARGIN)
PCOMR - ORIGIN MODE (CURSOR POSITION NUMBERS ARE
RELATIVE TO TOP LINE OF SCREEN)
PCSCNS - SCREEN MODE (NORMAL - LIGHT CHAR ON DARK BKGD)
PCSCNR - SCREEN MODE (REVERSE - LIGHT CHAR ON DARK BKGD)
PCSCLS - SET SMOOTH SCROLLING
PCSCLR - SET JUMP SCROLLING
----- - SET ANSI MODE (MUST BE DONE WITH ATS CONTROL
SEQUENCE)
PCANM - SET ATS (ALTERNATE TERMINAL SUPPORT) MODE
PCAWMS - SET AUTO WRAPAROUND
PCAWMR - RESET AUTO WRAPAROUND

REPORTS

DSR5 - STATUS REPORT REQUEST
DSR6 - CURSOR POSITION REPORT REQUEST
PCRTP0 - REQUEST TERMINAL PARAMETERS (SUBSEQUENT REPORTS
MAY BE UNSOLICITED)
PCRTP1 - REQUEST TERMINAL PARAMETERS (SUBSEQUENT REPORTS
ONLY IN RESPOSE TO HOST REQUEST)
DA - REQUEST DEVICE ATTRIBUTES

RESET TO INITIAL STATE
RIS - RESET TO INITIAL STATE
SCROLLING REGION
PCSTBM - SET TOP AND BOTTOM MARGINS
TABS
HTS - SET TAB
TBC0 - CLEAR TAB
TBC3 - CLEAR ALL TABS
TESTS
PCALN - FILL SCREEN WITH 'E'

SUBPROGRAMS REQUIRED
PART OF LANGUAGE
NONE
OTHERS
NONE

AUTHOR
STANLEY WILLNER - DTNSRDC CODE 1892.1

DATE WRITTEN: 03/07/84

DATE(S) REVISED
05/17/84 - CORRECT AND EXPAND CURSOR MOTION FIELDS

LOCATION OF DECKS
SOURCE DECK
UPDATE LIBRARY ON MSS: NSRDCPL,UN=CSYS
OBJECT DECK
EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT
BEGIN,DOCGET,,NSRDC,,VT100I,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'WARNING'

PURPOSE

FTN-CALLABLE 'WARNING' CONTROL CARD

FUNCTIONAL CATEGORIES: 01

LANGUAGE: FORTRAN IV EXTENDED

COMPUTERS (OPERATING SYSTEMS)

CDC 6000/CYBER 170 (NOS/BE)

REMARKS

NONE

USAGE

CALL WARNING (BANNER, OUTFILE)

DESCRIPTION OF PARAMETERS

BANNER - BANNER REQUEST. ONE OF:
"FOUO" - FOR OFFICIAL
USE ONLY
"OFFICIAL" - FOR OFFICIAL
USE ONLY
"PRIVACY" - PERSONAL DATA
PRIVACY ACT
OF 1974
"CONFIDENTIAL" - CONFIDENTIAL
"SECRET" - SECRET

NOTE: ONLY THE FIRST 7 CHARACTERS ARE TESTED.

OUTFILE - FORTRAN LOGICAL UNIT NUMBER OF THE OUTPUT FILE

CM REQUIRED: 1735B

OUTPUT DESCRIPTION

ONE BANNER PAGE WITH THE REQUESTED BANNER.

OUTPUT UNITS

UNIT #	LFN	USE
USER SPECIFIES		LISTABLE OUTPUT

SUBPROGRAMS REQUIRED
PART OF LANGUAGE
AND
OTHERS
NONE

ARITHMETIC STATEMENT FUNCTIONS
L71FMT - FAST L-FORMAT DECODE (LEFT-ADJ, ZERO-FILLED)

AUTHOR
DAVID V SOMMER - DTNSRDC CODE 1892.2

DATE WRITTEN: 09/19/79

DATE(S) REVISED
03/21/83 - RECOMPILE AT NOS/BE LEVEL 552

LOCATION OF DECKS
SOURCE
UPDATE LIBRARY ON MSS: NSRDCPL,UN=CSYS
OBJECT
EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT
BEGIN,DOCGET,,NSRDC,,WARNING,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'WEKDAY'

PURPOSE

DETERMINE THE DAY OF THE WEEK FOR ANY GREGORIAN DATE FROM
OCTOBER 15, 1582 THRU FEBRUARY 28, 4000

FUNCTIONAL CATEGORIES: M2

LANGUAGE: FORTRAN IV

COMPUTERS (OPERATING SYSTEMS)

CDC 6000/CYBER 170 (NOS/BE)

REMARKS

DATES FROM JANUARY 1, 1582 THRU OCTOBER 14, 1582 AND
AFTER FEBRUARY 28, 4000 THRU DECEMBER 31, 4000 ARE NOT
VALIDATED.

USAGE

CALL WEKDAY (IERR, IDAY, IGY, IGM, IGD)

DESCRIPTION OF PARAMETERS

IERR - RETURN CODE

0 - NO ERROR

1 - AT LEAST ONE OF IGY, IGM, IGD OUT OF RANGE

IDAY - WILL CONTAIN DAY-OF-WEEK

0 (SUNDAY) THRU 6 (SATURDAY)

IGY - GREGORIAN YEAR (EG, 1975)

IGM - GREGORIAN MONTH (1-12)

IGD - GREGORIAN DAY (1-31)

CM REQUIRED: 73B

METHOD

SEE IBM PROGRAM DESCRIPTION 360D 03.1.004

SUBPROGRAMS REQUIRED
PART OF LANGUAGE
MOD
OTHERS
NONE

AUTHOR
RICHARD CONNER - IBM

DATE WRITTEN: 10/15/66

DATE(S) REVISED
04/26/73 - REWRITTEN IN FORTRAN FOR CDC 6000 - DVS
04/25/79 - IMPLEMENTED ON BURROUGHS B7700 - DVS
03/21/83 - RECOMPILE AT NOS/BE LEVEL 552

LOCATION OF DECKS
SOURCE
UPDATE LIBRARY ON MSS: NSRDCPL,UD=CSYS
OBJECT
EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT
BEGIN,DOCGET,,NSRDC,,WEKDAY,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'ZBLANK'

PURPOSE

CHANGE BLANKS TO 00B AND VICE VERSA

FUNCTIONAL CATEGORIES: M4

LANGUAGE: FORTRAN IV EXTENDED

COMPUTERS (OPERATING SYSTEMS)

CDC 6000/CYBER 170 (NOS/BE)

REMARKS

NONE

USAGE

CALL ZBLANK (A, NA)

DESCRIPTION OF PARAMETERS

A - START OF AREA TO BE PROCESSED

NA - NUMBER OF WORDS TO BE PROCESSED

CM REQUIRED: 25B

SUBPROGRAMS REQUIRED

PART OF LANGUAGE

AND

OTHERS

NONE

AUTHOR

J. P. - KPS - NWL

DATE WRITTEN: 1973

DATE(S) REVISED

03/21/83 - RECOMPILE AT NOS/BE LEVEL 552

LOCATION OF DECKS

SOURCE

UPDATE LIBRARY ON MSS: NSRDCPL,UN=CSYS

OBJECT

EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC,,ZBLANK,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'ZEROFL'

PURPOSE

ZERO FIELD LENGTH (SECURITY EOJ)

FUNCTIONAL CATEGORIES: M4

LANGUAGE: FORTRAN IV

COMPUTERS (OPERATING SYSTEMS)

CDC 6000/CYBER 170 (NOS/BE)

REMARKS

'ZEROFL' ZEROS THE JOB'S FIELD LENGTH ABOVE 77B AND ENDS
THE JOB WITHOUT DAYFILE MESSAGES.

THE INTENDED USE IS AS THE TERMINATION ROUTINE, CALLED BY
REPRIEVE, WHENEVER A UTILITY PROGRAM HAS WITHIN ITS FIELD
LENGTH DATA THAT SHOULD NOT APPEAR IN A USER'S DUMP.

USAGE

CALL ZEROFL

CM REQUIRED: 21B

SUBPROGRAMS REQUIRED

PART OF LANGUAGE

NONE

OTHERS

NONE

AUTHOR

C FLINK - KP NWL

DATE WRITTEN: 08/73

DATE(S) REVISED

03/22/83 - RECOMPILE AT NOS/BE LEVEL 552

LOCATION OF DECKS

SOURCE

UPDATE LIBRARY ON MSS: NSRDCPL,UN=CSYS

OBJECT

EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC,,ZEROFL,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'ZEROS'
SUBROUTINE 'ZEROES'

PURPOSE

REPLACE BLANKS WITH (DISPLAY CODE) ZEROS, MULTIPLE FIELDS

FUNCTIONAL CATEGORIES: M4

LANGUAGE: CDC 6000 CP COMPASS

COMPUTERS (OPERATING SYSTEMS)
CDC 6000/CYBER 170 (NOS/BE)

REMARKS

'ZEROS' WILL REPLACE BLANKS WITH ZEROS UP TO THE 1ST
NON-BLANK CHARACTER IN A GIVEN FIELD.
IF THE 1ST NON-BLANK CHARACTER IS MINUS (-), THEN THAT
CHARACTER POSITION IS REPLACES WITH A ZERO AND THE 1ST
CHARACTER IN THE FIELD IS REPACED WITH A MINUS (-).

USAGE

CALL ZEROS (A, S1, L1, S2, L2, ..., SN, LN)
CALL ZEROES (A, S1, L1, S2, L2, ..., SN, LN)

DESCRIPTION OF PARAMETERS

A - ARRAY TO BE PROCESSED
S - STARTING BYTE OF A FIELD
(BYTE COUNT BEGINS WITH 1 FOR THE LEFTMOST BYTE IN 'A')
L - NUMBER OF BYTES IN THIS FIELD TO PROCESS

(UP TO 31 PAIRS OF SI,LI)

CM REQUIRED: 55B

SUBPROGRAMS REQUIRED

PART OF LANGUAGE
NONE
OTHERS
NONE

AUTHOR

T HERRING - KPS NWL

DATE WRITTEN: 12/09/70

DATE(S) REVISED

03/23/83 - RECOMPILE AT NOS/BE LEVEL 552

LOCATION OF DECKS

SOURCE
UPDATE LIBRARY ON MSS: NSRDCPL,UN=CSYS
OBJECT
EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC,,ZEROS,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'ZPFPUT'

PURPOSE

PUT USER-SPECIFIED PARAMETERS INTO ARRAY FOR LATER CALL TO ZPFUNC

FUNCTIONAL CATEGORIES: Q3

LANGUAGE: FORTRAN IV EXTENDED

COMPUTERS (OPERATING SYSTEMS)

CDC 6000/CYBER 170 (NOS/BE)

REMARKS

NO LONGER SUPPORTED SINCE SUBROUTINE 'ZPFUNC' HAS BEEN REPLACED BY SYSTEM SUBROUTINE 'PF'.

USAGE

CALL ZPFPUT (IPRMS, NW)
CALL ZPFPUT (IPRMS, NW, LFN, PFN, ID, TK, RD, EX, MD, CN,
MR, AC, CY, RP, XR, LC, RW, SN, VSN, FO, ST,
UV, RB)

FOR EXAMPLE:

CALL ZPFPUT (IPRMS, 0)
CALL ZPFPUT (IPRMS, 1, LFN)
CALL ZPFPUT (IPRMS, 5, LFN, PFN)
CALL ZPFPUT (IPRMS, 6, LFN, PFN, ID)
...
CALL ZPFPUT (IPRMS, 13, LFN, PFN, ID, TK, RD, EX, MD, CN,
MR, AC)
...
CALL ZPFPUT (IPRMS, 24, LFN, PFN, ID, TK, RD, EX, MD, CN,
MR, AC, CY, RP, XR, LC, RW, SN, VSN, FO, ST,
UV, RB)

DESCRIPTION OF PARAMETERS

IPRMS - ARRAY (MAXIMUM REQUIRED DIMENSION 24) TO BE
DEFINED
NW - 0 - SET ALL 24 WORDS TO ZERO
1 THRU 24 - DEFINE NW PARAMETERS FROM THE
FOLLOWING
LFN - LOCAL FILE NAME (1-7 CHARACTERS)
PFN - 4-WORD PERMANENT FILE NAME
ID - 1-9 CHARACTERS
TK - TURNKEY PASSWORD (1-9 CHARACTERS)
RD - READ PASSWORD (1-9 CHARACTERS)
EX - EXTEND PASSWORD (1-9 CHARACTERS)
MD - MODIFY PASSWORD (1-9 CHARACTERS)
CN - CONTROL PASSWORD (1-9 CHARACTERS)

MR - MULTIPLE-READ (0 OR NOT)
 AC - ACCOUNT NUMBER (10 CHARACTERS, LAST IS NUMERIC)
 CY - CYCLE (INTEGER -999 TO -1, 1 TO 999)
 RP - RETENTION PERIOD (0-999)
 XR - READ-ONLY PASSWORD (1-9 CHARACTERS)
 LC - LOWEST CYCLE (0 OR NOT)
 RW - MULTI-READ, SINGLE WRITE (0 OR NOT)
 SN - SETNAME (1-7 CHARACTERS)
 VSN - VOLUME SERIAL NUMBER (1-6 CHARACTERS,
 LEFT-JUSTIFIED). RESERVED FOR FUTURE.
 FO - FILE ORGANIZATION (2-CHARACTERS)
 ST - STATION ID (MULTI-FRAME)
 RESERVED FOR FUTURE.
 UV - UNIVERSAL PASSWORD (1-9 CHARACTERS)
 RB - PURGE RB CONFLICTS (0 OR NOT)

NOTE: ALL VARIABLES ARE TYPE INTEGER. CHARACTER DATA IS
 LEFT-JUSTIFIED AND MAY BE ZERO- OR BLANK-PADDED.
 TO CLEAR (OR OMIT) A SPECIFIC PARAMETER, USE 0.

CM REQUIRED: 137B

SUBPROGRAMS REQUIRED

PART OF LANGUAGE

MINO

OTHERS

NONE

AUTHOR

DAVID V SOMMER - DTNSRDC CODE 1892.2

DATE WRITTEN: 01/13/76

DATE(S) REVISED

01/20/76

09/23/80 - UPGRADE TO LEVEL 508 (UV AND RB ADDED)

03/23/83 - RECOMPILE AT NOS/BE LEVEL 552

LOCATION OF DECKS

SOURCE

UPDATE LIBRARY ON MSS: NSRDCPL,UN=CSYS

OBJECT

EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC,,ZPFPUT,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'ZPFUNC'

PURPOSE

CALLABLE PERMANENT FILE FUNCTIONS

FUNCTIONAL CATEGORIES: Q3

LANGUAGE: FORTRAN IV EXTENDED

COMPUTERS (OPERATING SYSTEMS)

CDC 6000/CYBER 170 (NOS/BE)

REMARKS

NO LONGER SUPPORTED SINCE SUBROUTINE 'ZPFUNC' HAS BEEN
REPLACED BY SYSTEM SUBROUTINE 'PF'.

USAGE

CALL ZPFUNC (IRC, IPRMS, NW)

DESCRIPTION OF PARAMETERS

IRC - INPUT: PERMANENT FILE FUNCTION DESIRED

- 1 - ATTACH (10B)
- 2 - CATALOG (20B)
- 3 - EXTEND (30B)
- 4 - PURGE (40B)
- 5 - RENAME (50B)
- 6 - PERM (60B)
- 24 - ALTER

IF THE VALUE IN PARENTHESES IS USED, THE 2-
OR 3-LINE SYSTEM MESSAGE WILL APPEAR IN THE
DAYFILE.

OUTPUT: ERROR RETURN CODE

(EITHER ZPFUNC- OR NOS/BE-GENERATED)

ZPFUNC-GENERATED

IRC	MEANING
-----	---------

---	-----
-----	-------

-1	IRC HAD ILLEGAL INPUT VALUE
----	-----------------------------

-2	LAST CHARACTER OF AC IS NOT DISPLAY CODE NUMERIC
----	---

NOS/BE-GENERATED: SEE NEXT PAGE

NOS/BE-GENERATED

DEC	OCT	COMND	MEANING
0	000	ALL	FUNCTION SUCCESSFUL
1	001		PFN/ID ERROR
2	002	A,P	LFN ALREADY IN USE
3	003	CEPR	UNKNOWN LFN
4	004	C	TOO MANY CYCLES (5 MAX)
5	005	C,E	PF CATALOG FULL
6	006		NO LFN OR PFN
8	010	C,E	LATEST INDEX NOT WRITTEN
9	011	C	FILE NOT ON A PF DEVICE
10	012	A	FILE NOT CATALOGED, SN=<SETNAME>
11	013	A	ARCHIVE RETRIEVAL ABORTED
12	014	C,R	BAD LPF COMMUNICATION
13	015	C	CY LIMIT REACHED (999 MAX)
14	016	C	PF DIRECTORY FULL
15	017	CEPR	FUNCTION ATTEMPTED ON A NON-PERMANENT FILE
16	020		FCN ATTEMPTED ON NON-LOCAL FILE
17	021	A	IMPROPER ARCHIVE RETRIEVAL CALL
18	022	C	FILE NEVER ASSIGN TO A DEVICE
19	023	A	CYCLE INCOMPLETE OR DUMPED
20	024	A	FILE ALREADY ATTACHED
21	025	A	FILE ARCHIVED
22	026		ILLEGAL CHARACTER IN FDB PARAM
23	027		ILLEGAL LFN
24	030	A	FILE DUMPED
25	031		ILLEGAL FUNCTION CODE
26	032	P	PURGE ATTEMPT IGNORED; USE RB PARAMETER
27	033		ALTER NEEDS EXCLUSIVE ACCESS
28	034		FDB IS TOO LARGE
29	035	C	FILE ALREADY IN SYSTEM
30	036	A	NO APF SPACE
31	037		PERMISSION CONFLICTS
32	040		ILLEGAL SETNAME SPECIFIED
33	041		DEVICE NOT MOUNTED AT CTL POINT
34	042		RBT CHAIN TOO LARGE FOR PFC
35	043	A,P	FILE RESIDES ON UNAVAILABLE DEVICE
36	044	A,P	FILE NOT AVAILABLE
56	070		PFM STOPPED BY SYSTEM
* 57	071		INCORRECT PERMISSION
* 58	072		FILE DEFINITION BLOCK ADDRESS INVALID (NOT RETURNED TO FDB)
* 59	073		I/O ERROR ON PFD/PFC READ/WRITE

* - ALWAYS CAUSES ABNORMAL JOB TERMINATION

IPRMS - PARAMETERS FOR PF FUNCTION
(UNUSED FIELDS MUST BE SET TO ZERO)

IPRMS	CONTENTS	FUNCTIONS	FORMAT
1	LFN	ALL	1-7 CHAR, LEFT* (IF 0, 1ST 7 CHAR OF PFN ARE USED (A,C,P))
2-5	PFN	A,C,P,R	1-40 CHAR, LEFT
6	ID	A,C,P,R	1-9 CHAR, LEFT
7	TK	**,***	1-9 CHAR, LEFT
8	RD	**,***	1-9 CHAR, LEFT
9	EX	**,***	1-9 CHAR, LEFT
10	MD	**,***	1-9 CHAR, LEFT
11	CN	**,***	1-9 CHAR, LEFT
12	MR	A,C	0 OR NOT
13	AC	C,R****	10 CHAR (LAST 3 NUMERIC)
14	CY	A,C,P,R	INTEGER (1-999) NEGATIVE TO RETURN VALUE
15	RP	C,R	INTEGER (0-999)
16	XR	C,R ****	1-9 CHAR, LEFT
17	LC	A,P	0 OR NOT
18	RW	A,C	0 OR NOT
19	SN	A,P	1-7 CHAR, LEFT
20	VSN		VOLUME SERIAL NUMBER (RESERVED FOR FUTURE)
21	FO	C	2-CHAR, LEFT (DA, IS, AK)
22	ST		STATION ID (MULTI-FRAME) (RESERVED FOR FUTURE)
23	UV	A,P	1-9 CHAR, LEFT
24	RB	P	0 OR NOT

A=ATTACH; C=CATALOG; P=PURGE; R=RENAME

- * LEFT=LEFT-JUSTIFIED, BLANK OR ZERO PADDED
- ** FOR A,P, INTERPRETED AS SUBMITTED PASSWORD
- FOR C, USED AS BOTH DEFINITION AND SUBMITTED PW
- *** FOR R, WHEN SET TO 1, THE PASSWORD IS CLEARED
- **** FOR C, WHEN OMITTED, AC IS TAKEN FROM CHARGE CARD
OR LOGIN

NW - NUMBER OF LAST FILLED ELEMENT IN IPRMS (OPTIONAL)

EXAMPLE

```
PROGRAM TEST (INPUT, OUTPUT,  
A          TAPE5=INPUT, TAPE6=OUTPUT)  
INTEGER IPRMS(24)  
DATA LFN   / 6LMYFILE/  
DATA ID    / 4LXXX/  
DATA IPFN1, IPFN2/ 10HPERMANENTF, 3LILE/  
DATA IAC   / 10H9876543210/      << SEE NOTE BELOW  
DATA IPW   / 8LPASSWORD/  
  
    . . .  
  
DO 10 I=1,13  
10 IPRMS(I) = 0  
   IPRMS( 1) = LFN  
   IPRMS( 2) = IPFN1  
   IPRMS( 3) = IPFN2  
   IPRMS( 6) = ID  
   IPRMS( 7) = IPW  
   IPRMS(13) = IAC      << SEE NOTE BELOW  
   IRC = 2  
   CALL ZPFUNC (IRC, IPRMS, 13)  
   IF (IRC .NE. 0) WRITE (6, 20) IRC, IRC  
20 FORMAT ('OERROR - IRC=', 03, 'B = ', 17)  
  
    . . .  
  
STOP  
END
```

THIS PROGRAM IS EQUIVALENT IN EFFECT TO THE FOLLOWING
CONTROL CARDS:

```
CATALOG(MYFILE,PERMANENTFILE,ID=XXXX,AC=9876543210,  
        PW=PASSWORD)
```

FOR A NEW CYCLE OF AN EXISTING FILE; OR

```
CATALOG(MYFILE,PERMANENTFILE,ID=XXXX,AC=9876543210,  
        TK=PASSWORD)
```

FOR THE CREATION OF A NEW FILE.

NOTE: IF THESE TWO LINES ARE OMITTED (THAT IS, AC IS
ZERO), AC WILL BE TAKEN FROM THE BATCH CHARGE CARD
OR THE INTERCOM LOGIN.

SUBPROGRAMS REQUIRED
PART OF LANGUAGE
AND SHIFT
OTHERS
IZPFBTZ
IZRT9ZR
NUMVAR
ZPFMAC
ZPFPSW

CM REQUIRED: 407B

AUTHOR
C M CHERNICK - DTNSRDC CODE 1832

DATE WRITTEN: 01/75

DATE(S) REVISED
05/75 01/02/76
09/23/80 - DVS - UPGRADE TO LEVEL 508 (ADD UV AND RB)
03/21/83 - RECOMPILE AT NOS/BE LEVEL 552

LOCATION OF DECKS
SOURCE
UPDATE LIBRARY ON MSS: NSRDCPL,UN=CSYS
OBJECT
EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT
BEGIN,DOCGET,,NSRDC,,ZPFUNC,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'ZRTPUT'

PURPOSE

PUT USER-SPECIFIED PARAMETERS INTO ARRAY FOR LATER CALL TO
ROUTE

FUNCTIONAL CATEGORIES: Q0

LANGUAGE: FORTRAN IV EXTENDED

COMPUTERS (OPERATING SYSTEMS)
CDC 6000/CYBER 170 (NOS/BE)

REMARKS

NO LONGER SUPPORTED SINCE SUBROUTINE 'ROUTE' HAS BEEN
REPLACED BY SYSTEM SUBROUTINE 'LF' (SEE SUBROUTINE 'PF'
DOCUMENT).

USAGE

CALL ZRTPUT (IPRMS, NW)
CALL ZRTPUT (IPRMS, NW, LFN, DC, TID, FID, DEF, RETJOB, FC,
EC, IC, STID, PRI, REP, NCD, SC)

FOR EXAMPLE:

CALL ZRTPUT (IPRMS, 0)
CALL ZRTPUT (IPRMS, 1, LFN)
CALL ZRTPUT (IPRMS, 2, LFN, DC)
...
CALL ZRTPUT (IPRMS, 14, LFN, DC, TID, FID, DEF, RETJOB, FC,
EC, IC, STID, PRI, REP, NCD, SC)

DESCRIPTION OF PARAMETERS

IPRMS - ARRAY (MAXIMUM REQUIRED DIMENSION 14) TO BE DEFINED
NW - 0 - SET ALL 14 WORDS TO ZERO
1 THRU 12 - DEFINE NW PARAMETERS FROM THE FOLLOWING
LFN - LOCAL FILE NAME (1-7 CHARACTERS)
DC - DISPOSITION CODE (2 CHARACTERS)
TID - TERMINAL IDENTIFICATION
1LC - CENTRAL SITE
2-CHARACTER TERMINAL ID
4LHERE - ROUTE TO THIS TERMINAL
FID - FILE IDENTIFICATION
1L* -OR-
1-5 CHARACTER FILE ID, PRECEDED BY *
DEF - 3LDEF - DEFER ROUTE UNTIL END OF JOB
RETJOB - NON-ZERO TO RETURN JOB NAME IN THIS WORD
FC - FORMS CODE (2 CHARACTERS)
EC - EXTERNAL CHARACTERISTICS
FOR PRINT:
2LB4, 2LB6, 2LA6, 2LA9
FOR PUNCH:
2LSB, 5L80COL, 3L026, 3L029, 5LASCII
IC - INTERNAL CHARACTERISTICS
0 OR 3LDIS - DISPLAY CODE
5LASCII - ASCII
3LBIN - BINARY

STID - 3-CHARACTER STATION ID
PRI - PRIORITY (TO ROUTING TERMINAL ONLY)
(0000B-7777B)
ALL OTHERS USE 0
REP - REPEAT COUNT (0-31 (37B))
NCD - 0 -OR-
1 - NO COMPLEMENTARY DAYFILE
(VALID ONLY IF IPRMS(5)=3LDEF)
SC - SPACING CODE FOR 580 PRINTER (0-77B)

NOTE: ALL VARIABLES ARE TYPE INTEGER. CHARACTER DATA IS
LEFT-JUSTIFIED AND ZERO-PADDED.
TO CLEAR (OR OMIT) A SPECIFIC PARAMETER, USE 0.

CM REQUIRED: 75B

SUBPROGRAMS REQUIRED
PART OF LANGUAGE
MINO
OTHERS
NONE

AUTHOR
DAVID V SOMMER - DTNSRDC CODE 1892.2

DATE WRITTEN: 01/19/76

DATE(S) REVISED
01/24/77 - ADD REP PARAMETER
11/30/77 - ADD NCD PARAMETER
04/07/83 - ADD SC PARAMETER

LOCATION OF DECKS
SOURCE
UPDATE LIBRARY ON MSS: NSRDCPL,UN=CSYS
OBJECT
EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT
BEGIN,DOCGET,,NSRDC,,ZRTPUT,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'ZSYSEQ'

PURPOSE

FORTRAN CALLABLE SYSTEM CALL

FUNCTIONAL CATEGORIES: Q3

LANGUAGE: CDC 6000 CP COMPASS

COMPUTERS (OPERATING SYSTEMS)
CDC 6000/CYBER 170 (NOS/BE)

REMARKS

NONE

USAGE

CALL ZSYSEQ (I)

DESCRIPTION OF PARAMETER

I - THE CONTENTS OF I ARE PUT INTO X6 BEFORE THE SYSTEM
IS CALLED

EXAMPLE

CALL SYSTEM ROUTINE DSP WITH PARAMETERS CONTAINED IN 'A':

CALL ZSYSEQ (4LDSPP .OR. LOCF(A))

NOTE: THE P AFTER DSP IS THE RECALL BIT. IF NO RECALL
REQUIRED, THEN:

CALL ZSYSEQ (3LDSP .OR. LOCF(A))

CM REQUIRED: 4B

SUBPROGRAMS REQUIRED

PART OF LANGUAGE

NONE

OTHERS

SYS=

AUTHOR

C M CHERNICK - DTNSRDC CODE 1832

DATE WRITTEN: 04/07/75

DATE(S) REVISED

03/23/83 - RECOMPILE AT NOS/BE LEVEL 552

LOCATION OF DECKS

SOURCE

UPDATE LIBRARY ON MSS: NSRDCPL,UN=CSYS

OBJECT

EDITLIB USER LIBRARY: NSRDC

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC,,ZSYSEQ,OUTPUT,MSACCES=<PASSWORD>.

***** SUBPROGRAM DOCUMENTATION (NSRDC5) *****

THIS CHAPTER CONTAINS THE MACHINE-READABLE DOCUMENTATION FOR THE SUBPROGRAMS IN LIBRARY 'NSRDC5'.

ALL DOCUMENT FILES RESIDE ON THE MASS STORAGE SYSTEM (MSS). YOUR MSACCES PASSWORD MUST BE SUBMITTED TO THE SYSTEM BEFORE DOCUMENTS CAN BE OBTAINED. THIS MAY BE DONE WITH A SEPARATE 'MSACCES' COMMAND OR BY USING THE MSACCES PARAMETER IN THE BEGIN STATEMENT.

*** HOW TO PRINT A DOCUMENT ***

INDIVIDUAL DOCUMENTS MAY BE PRINTED USING:

BEGIN,DOCGET,,NSRDC5,,<SUBPROG>,OUTPUT,MSACCES=<PASSWORD>.

WHERE <SUBPROG> IS THE DESIRED DOCUMENT.

SEVERAL DOCUMENTS MAY BE PRINTED AT ONE TIME USING:

BEGIN,DOCGET,,NSRDC5,,,OUTPUT,,,DOCS,MSACCES=<PASSWORD>.

WHERE DOCS IS A FILE CONTAINING THE NAMES OF THE DESIRED DOCUMENTS:

<SUBPROG1>,<SUBPROG2>,...,<SUBPROG1>
<SUBPROG1+1>,...,<SUBPROGN>

ALL DOCUMENTS MAY BE PRINTED USING:

BEGIN,DOCGET,,NSRDC5,,ALL,OUTPUT,MSACCES=<PASSWORD>.

TO PRINT THE DOCUMENT(S) ON THE XEROX 8700, EITHER:

A) ADD 'FID=<FID>' TO THE 'BEGIN,DOCGET,....'
WHERE <FID> IS THE FILE ID FOR THE BANNER

B) USE
BEGIN,XEROX,,OUTPUT,FID,,DOCPRT.

FUNCTION 'AC'

PURPOSE

GET ACCOUNT NUMBER FOR THIS JOB

FUNCTIONAL CATEGORIES: Q0

LANGUAGE: FORTRAN 77 EXTENDED

COMPUTERS (OPERATING SYSTEMS)

CDC 6000/CYBER 170 (NOS/BE)

REMARKS

'AC' MUST BE DECLARED CHARACTER*10 IN THE CALLING ROUTINE.

THE FTN4 VERSION HAS A SLIGHTLY DIFFERENT CALLING SEQUENCE.

USAGE

AC ()

DESCRIPTION OF PARAMETER

AC - CH*10 - WILL CONTAIN ACCOUNT NUMBER

CM REQUIRED: 52B

SUBPROGRAMS REQUIRED

PART OF LANGUAGE

AND OR SHIFT

OTHERS

RCPA - READ CONTROL POINT AREA

ARITHMETIC STATEMENT FUNCTIONS

L71FMT - FAST L-FORMAT DECODE (LEFT-ADJ, ZERO-FILLED)

METHOD

THE ACCOUNT NUMBER IS TAKEN FROM CONTROL POINT AREA.

AUTHOR

DAVID V SOMMER - DTNSRDC CODE 1892.2

DATE WRITTEN: 12/04/75

DATE(S) REVISED

02/27/76

12/10/81 - CONVERT TO FORTRAN 77 EXTENDED

LOCATION OF DECKS

SOURCE

UPDATE LIBRARY ON MSS: NSRDC5P, UN=CSYS

OBJECT

EDITLIB USER LIBRARY: NSRDC5

ANOTHER COPY OF THIS DOCUMENT

BEGIN, DOCGET, NSRDC5, AC, OUTPUT, MSACCES=<PASSWORD>.

SUBROUTINE 'ALTYM'

PURPOSE

OBTAIN CPA, CPB, CP, PP, IO AND WALL CLOCK TIMES SINCE
START OF JOB (OR INTERCOM SESSION)

FUNCTIONAL CATEGORIES: Q0 NO

LANGUAGE: FORTRAN 77 EXTENDED

COMPUTERS (OPERATING SYSTEMS)
CDC 6000/CYBER 170 (NOS/BE)

REMARKS
NONE

USAGE

CALL ALTYM (TIMES, CLOK)

DESCRIPTION OF PARAMETER

TIMES - REAL - 6-WORD ARRAY TO CONTAIN THE FOLLOWING:

- 1 - CPA TIME IN SECONDS
- 2 - CPB TIME IN SECONDS
- 3 - CP TIME IN SECONDS (CPA+CPB)
- 4 - PP TIME IN SECONDS
- 5 - IO TIME IN SECONDS
- 6 - WALL CLOCK TIME IN SECONDS

CLOK - CH*10 - WALL CLOCK (HH.MM.SS.)

CM REQUIRED: 102B

SUBPROGRAMS REQUIRED

PART OF LANGUAGE
NONE

OTHERS

HSM2S - CONVERT HH.MM.SS TO SECONDS
RCPA - READ CONTROL POINT AREA

ARITHMETIC STATEMENT FUNCTIONS

R65FMT - FAST R-FORMAT DECODE (RIGHT-ADJ, ZERO-FILLED)

AUTHOR

DAVID V SOMMER - DTNSRDC CODE 1892.2

DATE WRITTEN: 12/15/75 (ALTIME)

DATE(S) REVISED

07/05/83 - CONVERT TO FORTRAN 77
- CHANGE NAME FROM ALTIME TO ALTYM

LOCATION OF DECKS

SOURCE

UPDATE LIBRARY ON MSS: NSRDC5P,UN=CSYS

OBJECT

EDITLIB USER LIBRARY: NSRDC5

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC5,,ALTYM,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'BANR'

PURPOSE

PRINT A BANNER (LETTERS ARE 10 LINES HIGH, LINES ARE 131
PRINT POSITIONS LONG)

FUNCTIONAL CATEGORIES: J4

LANGUAGE: FORTRAN 77

COMPUTERS (OPERATING SYSTEMS)

CDC 6000/CYBER 170 (NOS/BE)

REMARKS

UPPER CASE ONLY (A-Z 0-9 + - * / () \$ = SPACE , . #
[] : " _ ! & ' ? < > @ \ ^ ;)

EACH BANNER REQUIRES 14 LINES (4 BLANKS, 10 FOR THE BANNER).
THUS, 3 BANNERS WILL FIT ON A PAGE AT 6 LINES PER INCH;
5 AT 8 LPI.

UP TO 10 CHARACTERS MAY APPEAR IN A BANNER. THE LINES ARE
131 PRINT POSITIONS LONG.

SEE SUBROUTINE 'BANR6'.

USAGE

CALL BANR (BANNER, IFILE, NEWPAG)

DESCRIPTION OF PARAMETERS

BANNER - 1-10 CHARACTERS TO BE PRINTED
(UP TO CHARACTER*10)

IFILE - NUMBER OF FILE ON WHICH BANNER IS TO BE WRITTEN

NEWPAG - ONE OF:

ZERO - BANNER IS WRITTEN ON NEW PAGE
NON-ZERO - BANNER IS WRITTEN ON SAME PAGE

CM REQUIRED: 1604B

OUTPUT UNITS

UNIT #	LFN/INT	USE
--------	---------	-----

USER SPECIFIES	LISTABLE OUTPUT
----------------	-----------------

EXAMPLES

PRINT THE BANNER 'HYSTERICAL' AT THE TOP OF THE NEXT PAGE
ON THE PRINTER FILE:

CALL BANR ('HYSTERICAL', L"OUTPUT", 0)

PRINT THE BANNER '10/19/77' ON THE SAME PAGE ON FILE 9:

CALL BANR ('10/19/77', 9, 1)

SUBPROGRAMS REQUIRED

PART OF LANGUAGE

LEN MIN

OTHERS

LASTCH - FIND LAST NON-BLANK

AUTHOR

DAVID V SOMMER - DTNSRDC CODE 1892.2

DATE WRITTEN: 02/18/75

DATE(S) REVISED

79/07/16 - RE-WRITTEN FOR B7700 (FORTRAN 66)

81/01/15 - CDC VERSION UPGRADED TO NOS/BE LEVEL 461

82/02/18 - CDC FORTRAN 77 VERSION WRITTEN

83/09/21 - SPEED UP BY USING INDEX FUNCTION

LOCATION OF DECKS

SOURCE

UPDATE LIBRARY ON MSS: NSRDC5P,UN=CSYS

OBJECT

EDITLIB USER LIBRARY: NSRDC5

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC5,,BANR,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'BANR6'

PURPOSE

PRINT A BANNER (LETTERS ARE 6 LINES HIGH, LINES ARE 80
PRINT POSITIONS LONG)

FUNCTIONAL CATEGORIES: J4

LANGUAGE: FORTRAN 77

COMPUTERS (OPERATING SYSTEMS)
CDC 6000/CYBER 170 (NOS/BE)

REMARKS

UPPER CASE ONLY (A-Z 0-9 + - * / () \$ = SPACE , . #
[] : " _ ! & ' ? < > @ \ ^ ;)

EACH BANNER REQUIRES 10 LINES (4 BLANKS, 6 FOR THE BANNER).
THUS, 6 BANNERS WILL FIT ON A PAGE AT 6 LINES PER INCH;
8 AT 8 LPI.

UP TO 10 CHARACTERS MAY APPEAR IN A BANNER. THE LINES ARE
80 PRINT POSITIONS LONG.

SEE SUBROUTINE 'BANR'.

USAGE

CALL BANR6 (BANNER, IFILE, NEWPAG)

DESCRIPTION OF PARAMETERS

BANNER - 1-10 CHARACTERS TO BE PRINTED
(UP TO CHARACTER*10)

IFILE - NUMBER OF FILE ON WHICH BANNER IS TO BE WRITTEN

NEWPAG - ONE OF:

ZERO - BANNER IS WRITTEN ON NEW PAGE
NON-ZERO - BANNER IS WRITTEN ON SAME PAGE

CM REQUIRED: 663B

OUTPUT UNITS

UNIT #	LFN/INT	USE
--------	---------	-----

USER SPECIFIES	LISTABLE OUTPUT
----------------	-----------------

EXAMPLES

PRINT THE BANNER 'HYSTERICAL' AT THE TOP OF THE NEXT PAGE
ON THE PRINTER FILE:

CALL BANR6 ('HYSTERICAL', L"OUTPUT", 0)

PRINT THE BANNER '10/19/77' ON THE SAME PAGE ON FILE 9:

CALL BANR6 ('10/19/77', 9, 1)

SUBPROGRAMS REQUIRED

PART OF LANGUAGE

LEN

OTHERS

LASTCH - FIND LAST NON-BLANK

AUTHOR

DAVID V SOMMER - DTNSRDC CODE 1892.2

DATE WRITTEN: 10/18/77

DATE(S) REVISED

79/07/16 - RE-WRITTEN FOR B7700 (FORTRAN 66)

81/01/15 - CDC VERSION UPGRADED TO NOS/BE LEVEL 461

82/02/19 - CDC FORTRAN 77 VERSION WRITTEN

83/09/21 - SPEED UP BY USING INDEX FUNCTION

LOCATION OF DECKS

SOURCE

UPDATE LIBRARY ON MSS: NSRDC5P,UN=CSYS

OBJECT

EDITLIB USER LIBRARY: NSRDC5

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC5,,BANR6,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'CENTER'

PURPOSE

CENTER A CHARACTER STRING

FUNCTIONAL CATEGORIES: M4

LANGUAGE: FORTRAN 77 EXTENDED

COMPUTERS (OPERATING SYSTEMS)

CDC 6000/CYBER 170 (NOS/BE)

REMARKS

SEE ALSO LEFT, RIGHT.

USAGE

CALL CENTER (CH, WORK)

DESCRIPTION OF PARAMETERS

CH - CHARACTER STRING TO BE CENTERED IN PLACE

WORK - CHARACTER VARIABLE THE SAME LENGTH AS 'CH'

CM REQUIRED: 114B

EXAMPLES

CHARACTER*20 A, WORK

DATA A/ 'ABCDEFGHIJ' /

...

CALL CENTER (A, WORK)

'A' NOW CONTAINS ' ABCDEFGHIJ '

SUBPROGRAMS REQUIRED

PART OF LANGUAGE

LEN

OTHERS

FIRSTCH - FIND FIRST NON-BLANK CHARACTER

LASTCH - FIND LAST NON-BLANK CHARACTER

AUTHOR

DAVID V SOMMER - DTNSRDC CODE 1892.2

DATE WRITTEN: 07/15/82

DATE(S) REVISED

LOCATION OF DECKS

SOURCE

UPDATE LIBRARY ON MSS: NSRDC5P, UN=CSYS

OBJECT

EDITLIB USER LIBRARY: NSRDC5

ANOTHER COPY OF THIS DOCUMENT

BEGIN, DOCGET, , NSRDC5, , CENTER, OUTPUT, MSACCES=<PASSWORD>.

FUNCTION 'CFIND'

PURPOSE

SCAN CHARACTER ARRAY FOR CHARACTER WORD

FUNCTIONAL CATEGORIES: M5

LANGUAGE: FORTRAN 77

COMPUTERS (OPERATING SYSTEMS)

CDC 6000/CYBER 170 (NOS/BE)

REMARKS

NONE

USAGE

NELT = CFIND (CARRAY, CMAX, CWORD)

DESCRIPTION OF PARAMETERS

CARRAY - CHAR - ARRAY TO BE SCANNED

CMAX - INT - NUMBER OF WORDS IN CARRAY TO BE SCANNED

CWORD - CHAR - WORD TO BE SCANNED FOR

CFIND - INT - WILL CONTAIN THE SUBSCRIPT OF THE ELEMENT
OF CARRAY WHICH MATCHES CWORD
(IF NO MATCH, CFIND = 0)

CM REQUIRED: 40B

EXAMPLE

```
...  
CHARACTER*1 LETTERS(26)  
DATA LETTERS/ 'A', 'B', ..., 'Z'/  
G = CFIND (LETTERS, 26, 'G')  
BAD = CFIND (LETTERS, 26, '"')  
PRINT *, 'G IS IN LETTERS', G, '  
PRINT *, '" IS IN LETTERS(', BAD, '  
...
```

WILL PRINT:

```
G IS IN LETTERS(7)  
" IS IN LETTERS(0)
```


SUBPROGRAMS REQUIRED

PART OF LANGUAGE

NONE

OTHERS

NONE

AUTHOR

DAVID V SOMMER - DTNSRDC CODE 1892.2

DATE WRITTEN: 12/14/81

DATE(S) REVISED

LOCATION OF DECKS

SOURCE

UPDATE LIBRARY ON MSS: NSRDC5P,UN=CSYS

OBJECT

EDITLIB USER LIBRARY: NSRDC5

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC5,,CFIND,OUTPUT,MSACCES=<PASSWORD>.

FUNCTION 'CHIN'

PURPOSE

CONVERT I-FORMATTED CHARACTER STRING TO INTEGER

FUNCTIONAL CATEGORIES: M2 M4

LANGUAGE: FORTRAN 77

COMPUTERS (OPERATING SYSTEMS)

CDC 6000/CYBER 170 (NOS/BE)

REMARKS

SEE CVCHIN FOR SUBROUTINE VERSION.

USAGE

CHIN (STRING)

DESCRIPTION OF PARAMETERS

STRING - INPUT CHARACTER STRING

CHIN - OUTPUT INTEGER

CM REQUIRED: 146B

DAYFILE ERROR MESSAGE

*** CHIN - CHARACTER <N> (<CH>) INVALID

A CHARACTER OTHER THAN A DIGIT, BLANK, PLUS, OR MINUS
WAS ENCOUNTERED IN THE CHARACTER STRING, OR, A PLUS OR
MINUS WAS ENCOUNTERED OUT OF PLACE. THE PROGRAM IS
ABORTED.

SUBPROGRAMS REQUIRED

PART OF LANGUAGE

INDEX LEN

OTHERS

ABORT - TERMINATE THE PROGRAM ABNORMALLY

AUTHORS

N L FICKEN - DTNSRDC CODE 1806

DAVID V SOMMER - DTNSRDC CODE 1892.2

DATE WRITTEN: 03/16/81 (NLF)

DATE(S) REVISED

03/30/83 - DVS - CHANGE NAME FROM CNVCHIN TO CVCHIN

07/06/83 - DVS - COMPLETE RE-WRITE TO MAKE MACHINE-
INDEPENDENT

- CONVERT FROM SUBROUTINE TO FUNCTION

- CHANGE NAME FROM CVCHIN TO CHIN

- MOVE ERROR MESSAGE FROM PRINTER TO DAYFILE

LOCATION OF DECKS

SOURCE

UPDATE LIBRARY ON MSS: NSRDC5P,UN=CSYS

OBJECT

EDITLIB USER LIBRARY: NSRDC5

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC5,,CHIN,OUTPUT,MSACCES=<PASSWORD>.

07/18/83

3-11

CHIN - 1 OF 1

SUBROUTINE 'CMMDUMP'

PURPOSE

DUMP COMMON MEMORY MANAGER (CMM) DYNAMIC AREA HEADERS AND TRAILER WITH OPTIONAL DUMP OF THE CONTENTS OF EACH BLOCK

FUNCTIONAL CATEGORIES: N2

LANGUAGE: FORTRAN 77 EXTENDED

COMPUTERS (OPERATING SYSTEMS)
CDC 6000/CYBER 170 (NOS/BE)

REMARKS

THIS SUBROUTINE PRINTS ON THE STANDARD OUTPUT FILE. IF THE FILE HAS NOT BEEN OPENED WHEN CMMDUMP IS CALLED, THE CONTENTS OF THE DYNAMIC AREA MAY CHANGE DURING DUMPING.

USAGE

CALL CMMDUMP (FULL)

DESCRIPTION OF PARAMETER

FULL - LOG - IN - TRUE - DUMP EACH BLOCK IN OCTAL AND CHARACTER
FALSE - DUMP ONLY THE HEADERS AND TRAIL

CM REQUIRED: 365B (+20B LBLD COMMON + 225B SUPPORTING DUMP)

OUTPUT DESCRIPTION

FOR FULL=.FALSE.:

EACH HEADER AND TRAILER WORD(S) IS LISTED IN OCTAL WITH ITS ADDRESS. POINTERS IN THE WORD ARE SHOWN SEPARATELY, AS ARE THE GROUP-ID (IF ANY) AND THE FWA/LWA SHRINK/GROW CHARACTERISTICS. FOR EACH BLOCK, THE NUMBER OF WORDS AND THE ADDRESS RANGE IS LISTED.

FOR FULL=.TRUE.:

SAME AS ABOVE. IN ADDITION, THE CONTENTS OF EACH BLOCK IS ALSO DUMPED IN OCTAL AND CHARACTER. MULTIPLE DUPLICATE LINES ARE REPLACED BY A SINGLE LINE WITH 'SAME'.

OUTPUT UNITS

UNIT #	LFN	USE
-----	-----	-----
	OUTPUT	LISTABLE OUTPUT

SUBPROGRAMS REQUIRED

PART OF LANGUAGE

AND CMMGOS SHIFT

PART OF PROGRAM

CMMDMPA - DUMP CONTENTS OF A BLOCK

OTHERS

LIBSYM - DUMMY ROUTINE TO FORCE LDSET,LIB=SYMLIB.

MFETCH - READ A MEMORY WORD

ARITHMETIC STATEMENT FUNCTIONS

BKD - EXTRACT BACKWARD POINTER

FLG - EXTRACT FLAG

FS - TEST FOR FWA SHRINK

FWD - EXTRACT FORWARD POINTER

GRID - EXTRACT GROUP-ID

HD2 - TEST FOR 2-WORD HEADER

LG - TEST FOR LWA GROW

LS - TEST FOR LWA SHRINK

AUTHOR

DAVID V SOMMER - DTNSRDC CODE 1892.2

DATE WRITTEN: 03/01/83

DATE(S) REVISED

04/14/83 - PRINT FIXED BLOCK AND FREE SPACE LENGTHS IN OCTAL
DECIMAL

LOCATION OF DECKS

SOURCE DECK

UPDATE LIBRARY ON MSS: NSRDC5P,UN=CSYS

OBJECT DECK

EDITLIB USER LIBRARY: NSRDC5

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC5,,CMMDUMP,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'CMMMERC'

PURPOSE

SUPPLY DESCRIPTION OF CMM MEMORY ERROR CODE

FUNCTIONAL CATEGORIES: QO

LANGUAGE: FORTRAN 77 EXTENDED

COMPUTERS (OPERATING SYSTEMS)

CDC 6000/CYBER 170 (NOS/BE)

REMARKS

THE DESCRIPTIONS ARE THOSE FOUND IN THE "COMMON MEMORY
MANAGER VERSION 1 REFERENCE MANUAL", 60499200 E, FIGURE B-1.

SEE ALSO 'CMMUERC'.

USAGE

CHARACTER MSG * 40

CALL CMMMERC (ME, MSG)

DESCRIPTION OF PARAMETERS

ME - THE MEMORY ERROR CODE
(THIS IS WORD 1 OF THE ARRAY FOR OWN-CODE
PROCESSING)

MSG - WILL CONTAIN TEXT DESCRIBING THE ERROR
(IF NO ERROR, 'NO MEMORY ERROR' IS RETURNED)

CM REQUIRED: 145B

SUBPROGRAMS REQUIRED

PART OF LANGUAGE

NONE

OTHERS

NONE

AUTHOR

DAVID V SOMMER - DTNSRDC CODE 1892.2

DATE WRITTEN: 02/23/83

DATE(S) REVISED

LOCATION OF DECKS

SOURCE DECK

UPDATE LIBRARY ON MSS: NSRDC5P, UN=CSYS

OBJECT DECK

EDITLIB USER LIBRARY: NSRDC5

ANOTHER COPY OF THIS DOCUMENT

BEGIN, DOCGET, , NSRDC5, , CMMMERC, OUTPUT, MSACCES=<PASSWORD>.

SUBROUTINE 'CMMOVEF'

PURPOSE

GET A LARGER AREA FROM CMM, MOVE OLD AREA TO NEW AREA,
RELEASE OLD AREA AND RESET POINTERS

FUNCTIONAL CATEGORIES: K2

LANGUAGE: FORTRAN 77 EXTENDED

COMPUTERS (OPERATING SYSTEMS)
CDC 6000/CYBER 170 (NOS/BE)

REMARKS

SINCE CYBER RECORD MANAGER (CRM - THE I/O SYSTEM) ALSO USES
CMM, IT IS NOT SAFE TO GET BLOCKS WHICH CAN GROW AT LWA.
THIS SUBROUTINE PROVIDES AN EASY WAY TO LENGTHEN AN AREA
WITHOUT THE RESTRICTIONS OF A GROW-AT-LWA BLOCK. THERE IS A
COST, HOWEVER. THERE WILL ALWAYS BE SOME DEAD SPACE IN THE
DYNAMIC AREA (WHICH CAN BE USED BY CMM FOR OTHER BLOCKS) AND
EACH TIME THE AREA IS "INCREASED", THE SUBROUTINE HAS TO
MOVE THE OLD AREA TO THE NEW AREA.

USAGE

CALL CMMOVEF (BASARY, FWA, PTR, NWORDS, INCR)

DESCRIPTION OF PARAMETERS

BASARY - DUMMY ARRAY BEING USED FOR THE DYNAMIC AREA
(THIS MUST BE THE FIRST WORD OF THE ACTUAL ARRAY,
NOT THE FIRST WORD OF THE DYNAMIC REFERENCE)
FWA - WILL CONTAIN THE FIRST WORD ADDRESS OF THE DYNAMIC
AREA (DO NOT CHANGE THIS!)
PTR - WILL POINT TO THE WORD BEFORE THE DYNAMIC AREA
(DO NOT CHANGE THIS!)
NWORDS - LENGTH OF CURRENT DYNAMIC AREA
INCR - NUMBER OF WORDS TO ADD TO THE DYNAMIC AREA

CM REQUIRED: 53B

EXAMPLE

USE SUBROUTINE CMMOVEF TO GET A 100-WORD ARRAY (10 WORDS AT
A TIME) AND FILL THE N-TH ELEMENT OF THE ARRAY WITH N:

```
INTEGER P(1)
...
FWA = 0
INCR = 10
MAX = 0
PTR = 0
DO 110 N=1,100
  IF (N .GT. MAX) THEN
    CALL CMMOVEF (P, FWA, PTR, N-1, INCR)
    MAX = MAX + INCR
  END IF
  P(PTR+N) = N
110 CONTINUE
```

METHOD

A NEW AREA OF LENGTH NWORDS+INCR IS OBTAINED USING CMMALF AND A NEW POINTER CALCULATED. IF THIS IS NOT THE FIRST CALL (THAT IS, IF FWA IS GREATER THAN 0), MOVE THE OLD AREA TO THE NEW AREA. FINALLY, SET FWA AND PTR TO POINT TO THE NEW AREA.

SUBPROGRAMS REQUIRED

PART OF LANGUAGE

LOCF

OTHERS

NONE

AUTHOR

DAVID V SOMMER - DTNSRDC CODE 1892.2

DATE WRITTEN: 02/22/83

DATE(S) REVISED

LOCATION OF DECKS

SOURCE DECK

UPDATE LIBRARY ON MSS: NSRDC5P,UN=CSYS

OBJECT DECK

EDITLIB USER LIBRARY: NSRDC5

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC5,,CMMOVEF,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'CMMPGFS'

PURPOSE

PRINT THE LARGEST BLOCK-SIZES AVAILABLE FOR ALL POSSIBLE
CONDITIONS

FUNCTIONAL CATEGORIES: NO

LANGUAGE: FORTRAN 77

COMPUTERS (OPERATING SYSTEMS)

CDC 6000/CYBER 170 (NOS/BE)

REMARKS

NONE

USAGE

CALL CMMPGFS

CM REQUIRED: 117B

OUTPUT DESCRIPTION

THE FOLLOWING EIGHT LINES ARE PRINTED AFTER A BLANK LINE:

INCREASE FL TO MAXLF	EXTEND BELOW HHA	LARGEST BLOCK-SIZE AVAILABLE	
0	0	<OCTAL>	<INTEGER>
1	1	<OCTAL>	<INTEGER>
1	0	<OCTAL>	<INTEGER>
1	1	<OCTAL>	<INTEGER>

OUTPUT UNITS

UNIT #	LFN	USE
-----	-----	-----
	OUTPUT	LISTABLE OUTPUT

SUBPROGRAMS REQUIRED

PART OF LANGUAGE

CMMGFS

OTHERS

NONE

AUTHOR

DAVID V SOMMER - DTNSRDC CODE 1892.2

DATE WRITTEN: 03/02/83

DATE(S) REVISED

04/14/83 - PRINT IN OCTAL AND DECIMAL

LOCATION OF DECKS

SOURCE DECK

UPDATE LIBRARY ON MSS: NSRDC5P,UN=CSYS

OBJECT DECK

EDITLIB USER LIBRARY: NSRDC5

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC5,,CMMPGFS,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'CMMPGOS'

PURPOSE

PRINT THE CONTENTS OF THE ARRAY RETURNED BY SUBROUTINE
CMMGOS

FUNCTIONAL CATEGORIES: NO

LANGUAGE: FORTRAN 77

COMPUTERS (OPERATING SYSTEMS)
CDC 6000/CYBER 170 (NOS/BE)

REMARKS

YOU MUST CALL CMMGOS BEFORE CALLING THIS SUBROUTINE.

USAGE

CALL CMMPGOS (GOS)

DESCRIPTION OF PARAMETER

GOS - THE 16-WORD ARRAY RETURNED BY AN EARLIER CALL TO
SUBROUTINE CMMGOS

CM REQUIRED: 17B

OUTPUT DESCRIPTION

THE FOLLOWING TEN LINES ARE PRINTED AFTER A BLANK LINE:

DABA	=	<OCTAL>	<INTEGER>
HHA	=	<OCTAL>	<INTEGER>
FL	=	<OCTAL>	<INTEGER>
MAXFL	=	<OCTAL>	<INTEGER>
PCT OF MEMORY USED	=	<OCTAL>	<INTEGER>
# CONTIG FREE WORDS	=	<OCTAL>	<INTEGER>
# FXD-POS BLOCKS	=	<OCTAL>	<INTEGER>
# ALLOC FXD-POS WORDS	=	<OCTAL>	<INTEGER>
# FREE AREAS	=	<OCTAL>	<INTEGER>
# FREE WORDS	=	<OCTAL>	<INTEGER>

OUTPUT UNITS

UNIT #	LFN	USE
-----	-----	-----
	OUTPUT	LISTABLE OUTPUT

EXAMPLE

```
INTEGER GOS(0:15)
...
CALL CMMGOS (GOS)
CALL CMMPGOS (GOS)
...
```

SUBPROGRAMS REQUIRED

PART OF LANGUAGE
NONE
OTHERS
NONE

AUTHOR

DAVID V SOMMER - DTNSRDC CODE 1892.2

DATE WRITTEN: 03/02/83

DATE(S) REVISED

LOCATION OF DECKS

SOURCE DECK

UPDATE LIBRARY ON MSS: NSRDC5P,UN=CSYS

OBJECT DECK

EDITLIB USER LIBRARY: NSRDC5

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC5,,CMMPGOS,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'CMPGSS'

PURPOSE

PRINT THE CONTENTS OF THE ARRAY RETURNED BY SUBROUTINE
CMMGSS

FUNCTIONAL CATEGORIES: NO

LANGUAGE: FORTRAN 77

COMPUTERS (OPERATING SYSTEMS)
CDC 6000/CYBER 170 (NOS/BE)

REMARKS

YOU MUST CALL CMMGSS BEFORE CALLING THIS SUBROUTINE.

USAGE

CALL CMPGSS (GSS)

DESCRIPTION OF PARAMETER

GSS - THE 6-WORD ARRAY RETURNED BY AN EARLIER CALL TO
SUBROUTINE CMMGSS

CM REQUIRED: 62B

OUTPUT DESCRIPTION

THE FOLLOWING FOUR LINES ARE PRINTED AFTER A BLANK LINE:

MAX # ALLOC WORDS = <OCTAL> <INTEGER>
MAX FL ATTAINED = <OCTAL> <INTEGER>
FL INCREASES = <OCTAL> <INTEGER>
FL DECREASES = <OCTAL> <INTEGER>

OUTPUT UNITS

UNIT #	LFN	USE
-----	-----	-----
	OUTPUT	LISTABLE OUTPUT

EXAMPLE

INTEGER GSS(0:5)
...
CALL CMMGSS (GSS)
CALL CMPGSS (GSS)
...

SUBPROGRAMS REQUIRED

PART OF LANGUAGE
NONE
OTHERS
NONE

AUTHOR

DAVID V SOMMER - DTNSRDC CODE 1892.2

DATE WRITTEN: 03/02/83

DATE(S) REVISED

LOCATION OF DECKS

SOURCE DECK

UPDATE LIBRARY ON MSS: NSRDC5P,UN=CSYS

OBJECT DECK

EDITLIB USER LIBRARY: NSRDC5

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC5,,CMMPGSS,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'CMMUERC'

PURPOSE

SUPPLY DESCRIPTION OF CMM USER ERROR CODE

FUNCTIONAL CATEGORIES: Q0

LANGUAGE: FORTRAN 77 EXTENDED

COMPUTERS (OPERATING SYSTEMS)

CDC 6000/CYBER 170 (NOS/BE)

REMARKS

THE DESCRIPTIONS ARE THOSE FOUND IN THE "COMMON MEMORY
MANAGER VERSION 1 REFERENCE MANUAL", 60499200 E, TABLE B-1.

SEE ALSO 'CMMMERC'.

USAGE

CHARACTER MSG * 40

CALL CMMUERC (UE, MSG)

DESCRIPTION OF PARAMETERS

UE - THE USER ERROR CODE
(THIS IS WORD 2 OF THE ARRAY FOR OWN-CODE
PROCESSING)

MSG - WILL CONTAIN TEXT DESCRIBING THE ERROR
(IF NO ERROR, 'NO ERROR' IS RETURNED)

CM REQUIRED: 502B

SUBPROGRAMS REQUIRED

PART OF LANGUAGE

NONE

OTHERS

NONE

AUTHOR

DAVID V SOMMER - DTNSRDC CODE 1892.2

DATE WRITTEN: 02/23/83

DATE(S) REVISED

LOCATION OF DECKS

SOURCE DECK

UPDATE LIBRARY ON MSS: NSRDC5P,UN=CSYS

OBJECT DECK

EDITLIB USER LIBRARY: NSRDC5

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC5,,CMMUERC,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'CSHUFL'

PURPOSE

SHUFFLE A CHARACTER ARRAY

FUNCTIONAL CATEGORIES: M1

LANGUAGE: FORTRAN 77 EXTENDED

COMPUTERS (OPERATING SYSTEMS)

CDC 6000/CYBER 170 (NOS/BE)

REMARKS

NONE

USAGE

CALL CSHUFL (ORIG, NELTS, REORDR, SUBARY, WORK)

DESCRIPTION OF PARAMETERS

ORIG - CHAR - ORIGINAL ARRAY TO BE SHUFFLED

NELTS - INT - NUMBER OF ELEMENTS TO BE SHUFFLED

REORDR - CHAR - SHUFFLED ARRAY

SUBARY - INT - ARRAY TO CONTAIN THE REORDERED SUBSCRIPTS
(THE ORIGINAL POSITION OF REORDR(I) IS
ORIG(SUBARY(I)))

WORK - CHAR*1 - WORK ARRAY

CM REQUIRED: 144B

METHOD

THE RANDOM NUMBER GENERATOR IS STARTED USING THE TIME SINCE
THE BEGINNING OF THE JOB, SO EACH EXECUTION SHOULD PRODUCE A
DIFFERENT RE-ORDERING.

SUBPROGRAMS REQUIRED

PART OF LANGUAGE

RANF RANSET SECOND

OTHERS

NONE

AUTHOR

DAVID V SOMMER - DTNSRDC CODE 1892.2

DATE WRITTEN: 12/16/81

DATE(S) REVISED

LOCATION OF DECKS

SOURCE

UPDATE LIBRARY ON MSS: NSRDC5P, UN=CSYS

OBJECT

EDITLIB USER LIBRARY: NSRDC5

ANOTHER COPY OF THIS DOCUMENT

BEGIN, DOCGET, , NSRDC5, , CSHUFL, OUTPUT, MSACCES=<PASSWORD>.

SUBROUTINE 'CSORT'

PURPOSE

SORT A CHARACTER ARRAY

FUNCTIONAL CATEGORIES: M1

LANGUAGE: FORTRAN 77

COMPUTERS (OPERATING SYSTEMS)

CDC 6000/CYBER 170 (NOS/BE)

REMARKS

SEE CSORTD, CSORTN, CSORT2.

USAGE

CALL CSORT (CARRAY, NELTS, CTEMP)

DESCRIPTION OF PARAMETERS

CARRAY - CHAR - ARRAY TO BE SORTED

NELTS - INT - NUMBER OF ELEMENTS TO BE SORTED

CTEMP - CHAR - VARIABLE OF THE SAME LENGTH AS 'CARRAY'
USED FOR SWAPPING

CM REQUIRED: 110B

SUBPROGRAMS REQUIRED

PART OF LANGUAGE

NONE

OTHERS

NONE

AUTHOR

DAVID V SOMMER - DTNSRDC CODE 1892.2

DATE WRITTEN: 12/17/81

DATE(S) REVISED

02/09/82 - RESTRUCTURE

LOCATION OF DECKS

SOURCE

UPDATE LIBRARY ON MSS: NSRDC5P, UN=CSYS

OBJECT

EDITLIB USER LIBRARY: NSRDC5

ANOTHER COPY OF THIS DOCUMENT

BEGIN, DOCGET, , NSRDC5, , CSORT, OUTPUT, MSACCES=<PASSWORD>.

SUBROUTINE 'CSORTD'

PURPOSE

 SORT A CHARACTER ARRAY (DESCENDING)

FUNCTIONAL CATEGORIES: M1

LANGUAGE: FORTRAN 77

COMPUTERS (OPERATING SYSTEMS)

 CDC 6000/CYBER 170 (NOS/BE)

REMARKS

 SEE CSORT, CSORTN, CSORT2.

USAGE

 CALL CSORTD (CARRAY, NELTS, CTEMP)

DESCRIPTION OF PARAMETERS

 CARRAY - CHAR - ARRAY TO BE SORTED

 NELTS - INT - NUMBER OF ELEMENTS TO BE SORTED

 CTEMP - CHAR - VARIABLE OF THE SAME LENGTH AS 'CARRAY'
 USED FOR SWAPPING

CM REQUIRED: 110B

SUBPROGRAMS REQUIRED

 PART OF LANGUAGE

 NONE

 OTHERS

 NONE

AUTHOR

 DAVID V SOMMER - DTNSRDC CODE 1892.2

DATE WRITTEN: 02/09/82

DATE(S) REVISED

LOCATION OF DECKS

 SOURCE

 UPDATE LIBRARY ON MSS: NSRDC5P,UN=CSYS

 OBJECT

 EDITLIB USER LIBRARY: NSRDC5

ANOTHER COPY OF THIS DOCUMENT

 BEGIN,DOCGET,,NSRDC5,,CSORTD,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'CSORTN'

PURPOSE

SORT A CHARACTER ARRAY (HAVING AN ASSOCIATED NON-CHARACTER
ARRAY)

FUNCTIONAL CATEGORIES: M1

LANGUAGE: FORTRAN 77

COMPUTERS (OPERATING SYSTEMS)
CDC 6000/CYBER 170 (NOS/BE)

REMARKS

SEE CSORT, CSORTD, CSORT2.

USAGE

CALL CSORTN (CARRAY, NELTS, CTEMP, ASSOC)

DESCRIPTION OF PARAMETERS

CARRAY - CHAR - ARRAY TO BE SORTED
NELTS - INT - NUMBER OF ELEMENTS TO BE SORTED
CTEMP - CHAR - VARIABLE OF THE SAME LENGTH AS 'CARRAY'
USED FOR SWAPPING
ASSOC - - ASSOCIATED NON-CHARACTER ARRAY WHICH WILL
BE RE-ORDERED TO MAINTAIN A 1-TO-1 CORRE-
SPONDENCE WITH THE ELEMENTS OF 'CARRAY'

CM REQUIRED: 117B

SUBPROGRAMS REQUIRED

PART OF LANGUAGE
NONE
OTHERS
NONE

AUTHOR

DAVID V SOMMER - DTNSRDC CODE 1892.2

DATE WRITTEN: 12/17/81

DATE(S) REVISED

02/09/82 - RESTRUCTURE

LOCATION OF DECKS

SOURCE
UPDATE LIBRARY ON MSS: NSRDC5P, UN=CSYS
OBJECT
EDITLIB USER LIBRARY: NSRDC5

ANOTHER COPY OF THIS DOCUMENT

BEGIN, DOCGET, , NSRDC5, , CSORTN, OUTPUT, MSACCES=<PASSWORD>.

SUBROUTINE 'CSORT2'

PURPOSE

SORT A CHARACTER ARRAY (HAVING AN ASSOCIATED CHARACTER
ARRAY)

FUNCTIONAL CATEGORIES: M1

LANGUAGE: FORTRAN 77

COMPUTERS (OPERATING SYSTEMS)
CDC 6000/CYBER 170 (NOS/BE)

REMARKS

SEE CSORT, CSORTD, CSORTN.

USAGE

CALL CSORT2 (CARRAY, NELTS, CTEMP, CASSOC, CTEMPA)

DESCRIPTION OF PARAMETERS

CARRAY - CHAR - ARRAY TO BE SORTED
NELTS - INT - NUMBER OF ELEMENTS TO BE SORTED
CTEMP - CHAR - VARIABLE OF THE SAME LENGTH AS 'CARRAY'
USED FOR SWAPPING
CASSOC - CHAR - ASSOCIATED ARRAY WHICH WILL BE RE-ORDERED
TO MAINTAIN A 1-TO-1 CORRESPONDENCE WITH
THE ELEMENTS OF 'CARRAY'
CTEMPA - CHAR - VARIABLE OF THE SAME LENGTH AS 'CASSOC'
USED FOR SWAPPING

CM REQUIRED: 151B

SUBPROGRAMS REQUIRED

PART OF LANGUAGE

NONE

OTHERS

NONE

AUTHOR

DAVID V SOMMER - DTNSRDC CODE 1892.2

DATE WRITTEN: 12/17/81

DATE(S) REVISED

02/09/82 - RESTRUCTURE

LOCATION OF DECKS

SOURCE

UPDATE LIBRARY ON MSS: NSRDC5P, UN=CSYS

OBJECT

EDITLIB USER LIBRARY: NSRDC5

ANOTHER COPY OF THIS DOCUMENT

BEGIN, DOCGET, , NSRDC5, , CSORT2, OUTPUT, MSACCES=<PASSWORD>.

SUBROUTINE 'CVCHIN'

PURPOSE

CONVERT I-FORMATTED CHARACTER STRING TO INTEGER

FUNCTIONAL CATEGORIES: M2 M4

LANGUAGE: FORTRAN 77 EXTENDED

COMPUTERS (OPERATING SYSTEMS)

CDC 6000/CYBER 170 (NOS/BE)

REMARKS

NONE

USAGE

CALL CVCHIN (CHAR, INT)

DESCRIPTION OF PARAMETERS

CHAR - INPUT CHARACTER STRING

INT - OUTPUT INTEGER

CM REQUIRED: 130B

OUTPUT UNITS

UNIT	#	LFN	USE
------	---	-----	-----

		OUTPUT	ERROR MESSAGE
--	--	--------	---------------

ERROR MESSAGES

*** CVCHIN - INVALID CHARACTER IN STRING = <CHAR>

A CHARACTER OTHER THAN A DIGIT, BLANK, PLUS, OR MINUS
WAS ENCOUNTERED IN THE CHARACTER STRING. THE PROGRAM
IS ABORTED.

SUBPROGRAMS REQUIRED

PART OF LANGUAGE

ICHAR LEN

OTHERS

ABORT - TERMINATE THE PROGRAM ABNORMALLY

AUTHOR

N L FICKEN - DTNSRDC CODE 1806

DATE WRITTEN: 03/16/81

DATE(S) REVISED

03/30/83 - NAME CHANGED FROM CNVCHIN TO CVCHIN

LOCATION OF DECKS

SOURCE

UPDATE LIBRARY ON MSS: NSRDC5P,UN=CSYS

OBJECT

EDITLIB USER LIBRARY: NSRDC5

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC5,,CVCHIN,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'CVCHOL'

PURPOSE

CONVERT CHARACTER STRING TO HOLLERITH STRING

FUNCTIONAL CATEGORIES: M2 M4

LANGUAGE: FORTRAN 77 EXTENDED

COMPUTERS (OPERATING SYSTEMS)

CDC 6000/CYBER 170 (NOS/BE)

REMARKS

NONE

USAGE

CALL CVCHOL (CHAR, HOL, LENGTH)

DESCRIPTION OF PARAMETERS

CHAR - INPUT CHARACTER STRING

HOL - OUTPUT HOLLERITH STRING

LENGTH - LENGTH OF STRING

CM REQUIRED: 71B

SUBPROGRAMS REQUIRED

PART OF LANGUAGE

ICHAR LEN OR SHIFT

OTHERS

NONE

AUTHOR

N L FICKEN - DTNSRDC CODE 1806

DATE WRITTEN: 03/16/81

DATE(S) REVISED

03/30/83 - NAME CHANGED FROM CNVCHOL TO CVCHOL

LOCATION OF DECKS

SOURCE

UPDATE LIBRARY ON MSS: NSRDC5P,UN=CSYS

OBJECT

EDITLIB USER LIBRARY: NSRDC5

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC5,,CVCHOL,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'CVHOCH'

PURPOSE

CONVERT HOLLERITH STRING TO CHARACTER STRING

FUNCTIONAL CATEGORIES: M2 M4

LANGUAGE: FORTRAN 77 EXTENDED

COMPUTERS (OPERATING SYSTEMS)

CDC 6000/CYBER 170 (NOS/BE)

REMARKS

NONE

USAGE

CALL CVHOCH (CHAR, HOL, LENGTH)

DESCRIPTION OF PARAMETERS

CHAR - OUTPUT CHARACTER STRING

HOL - INPUT HOLLERITH STRING

LENGTH - LENGTH OF STRING

CM REQUIRED: 123B

SUBPROGRAMS REQUIRED

PART OF LANGUAGE

AND

ICHAR

LEN

SHIFT

OTHERS

NONE

AUTHOR

N L FICKEN - DTNSRDC CODE 1806

DATE WRITTEN: 03/16/81

DATE(S) REVISED

03/30/83 - NAME CHANGED FROM CNVHOCH TO CVHOCH

LOCATION OF DECKS

SOURCE

UPDATE LIBRARY ON MSS: NSRDC5P,UN=CSYS

OBJECT

EDITLIB USER LIBRARY: NSRDC5

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC5,,CVHOCH,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'CVINCH'

PURPOSE

CONVERT INTEGER TO CHARACTER STRING

FUNCTIONAL CATEGORIES: M2 M4

LANGUAGE: FORTRAN 77 EXTENDED

COMPUTERS (OPERATING SYSTEMS)

CDC 6000/CYBER 170 (NOS/BE)

REMARKS

IF LEN(CHAR) < LENGTH, THE OUTPUT STRING WILL HAVE ONLY
LEN(CHAR) CHARACTERS.

AN ASTERISK WILL BE INSERTED INTO THE LEFTMOST CHARACTER
POSITION IF THE OUTPUT STRING IS TOO SMALL TO HOLD THE
ENTIRE NUMBER.

USAGE

CALL CVINCH (CHAR, INT, LENGTH)

DESCRIPTION OF PARAMETERS

CHAR - OUTPUT CHARACTER STRING OR SUBSTRING
INT - INPUT INTEGER VALUE
LENGTH - LENGTH OF OUTPUT SUBSTRING IN CHAR

CM REQUIRED: 164B

SUBPROGRAMS REQUIRED

PART OF LANGUAGE

CHAR IABS LEN

OTHERS

NONE

AUTHOR

N L FICKEN - DTNSRDC CODE 1806

DATE WRITTEN: 03/16/81

DATE(S) REVISED

03/30/83 - NAME CHANGED FROM CNVINCH TO CVINCH

LOCATION OF DECKS

SOURCE

UPDATE LIBRARY ON MSS: NSRDC5P,UN=CSYS

OBJECT

EDITLIB USER LIBRARY: NSRDC5

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC5,,CVINCH,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'DMPCPA'

PURPOSE

SHORT DUMP OF JOB CONTROL POINT AREA

FUNCTIONAL CATEGORIES: N2

LANGUAGE: FORTRAN 77 EXTENDED

COMPUTERS (OPERATING SYSTEMS)

CDC 6000/CYBER 170 (NOS/BE)

REMARKS

FOR A FULL, ANNOTATED DUMP, CALL DUMPCPA.

USAGE

CALL DMPCPA

CM REQUIRED: 372B

OUTPUT DESCRIPTION

AN OCTAL AND CHARACTER DUMP OF THE 200 (OCTAL) WORDS OF THE
CONTROL POINT AREA.

OUTPUT UNITS

UNIT #	LFN	USE
-----	-----	-----
	OUTPUT	LISTABLE OUTPUT

SUBPROGRAMS REQUIRED

PART OF LANGUAGE

DATE TIME

OTHERS

RCPA - READ CONTROL POINT AREA

AUTHOR

DAVID V SOMMER - DTNSRDC CODE 1892.2

DATE WRITTEN: 12/29/75

DATE(S) REVISED

05/25/83 - ORIGINAL SOURCE LOST - RE-WRITTEN IN FORTRAN 77

LOCATION OF DECKS

SOURCE

UPDATE LIBRARY ON MSS: NSRDC5P,UN=CSYS

OBJECT

EDITLIB USER LIBRARY: NSRDC5

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC5,,DMPCPA,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'DUMPXPK'

PURPOSE

DUMP EXCHANGE PACKAGE (REGISTERS, POINTERS, ETC.)

FUNCTIONAL CATEGORIES: N2

LANGUAGE: FORTRAN 77 EXTENDED

REMARKS

PRINTOUT IS AT 8 LINES PER INCH AND IS RESTORED TO 6 LINES
PER INCH BEFORE RETURN.

USAGE

CALL DUMPXPK

CM REQUIRED: 711B

OUTPUT UNITS

UNIT #	LFN	USE
-----	-----	-----
	OUTPUT	LISTABLE OUTPUT

SUBPROGRAMS REQUIRED

PART OF LANGUAGE

AND MASK SHIFT

OTHERS

MFETCH - READ WORD OF MEMORY

MFRAME - GET EXECUTING MAINFRAME

RCPA - READ CONTROL POINT AREA

ARITHMETIC STATEMENT FUNCTIONS

FAST L-FORMAT DECODE (LEFT-ADJ, ZERO-FILLED)

L32FMT L35FMT L38FMT

FAST L-FORMAT DECODE (SHIFTED -42)

S32FMT S35FMT S38FMT

FAST L-FORMAT DECODE (SHIFTED 42, -42)

X38FMT

FAST R-FORMAT DECODE (RIGHT-ADJ, ZERO-FILLED)

R21FMT R23FMT R25FMT R27FMT R29FMT

R32FMT R35FMT

OTHERS

CY176 - TEST FOR CYBER 176

INRANGE - TEST FOR ADDRESS WITHIN FL

AUTHOR

DAVID V SOMMER - DTNSRDC CODE 1892.2

DATE WRITTEN: 05/04/83

DATE(S) REVISED

LOCATION OF DECKS

SOURCE

UPDATE LIBRARY ON MSS: NSRDC5P,UN=CSYS

OBJECT

EDITLIB USER LIBRARY: NSRDC5

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC5,,DUMPPXPK,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'ELTYM'

PURPOSE

OBTAIN CPA, CPB, CP, PP, IO AND WALL CLOCK TIMES SINCE
LAST CALL

FUNCTIONAL CATEGORIES: Q0

LANGUAGE: FORTRAN 77 EXTENDED

COMPUTERS (OPERATING SYSTEMS)
CDC 6000/CYBER 170 (NOS/BE)

REMARKS
NONE

USAGE
CALL ELTYM (TIMES, CLOK)

DESCRIPTION OF PARAMETER

TIMES - 6-WORD REAL ARRAY TO CONTAIN THE FOLLOWING:

- 1 - CPA TIME IN SECONDS
- 2 - CPB TIME IN SECONDS
- 3 - CP TIME IN SECONDS (CPA+CPB)
- 4 - PP TIME IN SECONDS
- 5 - IO TIME IN SECONDS
- 6 - WALL CLOCK TIME IN SECONDS

CLOK - CH*10 - WALL CLOCK (HH.MM.SS.)

CM REQUIRED: 137B

SUBPROGRAMS REQUIRED

PART OF LANGUAGE

NONE

OTHERS

- HMS2S - CONVERT HH.MM.SS TO SECONDS
- RCPA - READ CONTROL POINT AREA
- S2HMS - CONVERT SECONDS TO ' HH.MM.SS. '

ARITHMETIC STATEMENT FUNCTIONS

R65FMT - FAST R-FORMAT DECODE (RIGHT-ADJ, ZERO-FILLED)

AUTHOR

DAVID V SOMMER - DTNSRDC CODE 1892.2

DATE WRITTEN: 12/15/75 (ELTIME)

DATE(S) REVISED

- 10/31/77 - ADJUST FOR MIDNIGHT
- 03/18/83 - CONVERT TO FORTRAN 77
- CHANGE NAME FROM ELTIME TO ELTYM

LOCATION OF DECKS

SOURCE

UPDATE LIBRARY ON MSS: NSRDC5P,UN=CSYS

OBJECT

EDITLIB USER LIBRARY: NSRDC5

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC5,,ELTYM,OUTPUT,MSACCES=<PASSWORD>.

✓ AD-A148 792

COMPUTER CENTER CDC LIBRARIES/NSRD (SUBPROGRAMS)(U)
DAVID W TAYLOR NAVAL SHIP RESEARCH AND DEVELOPMENT
CENTER BET. D V SOMMER ET AL. JUN 84
DTNSRDC/CMLD-84-12

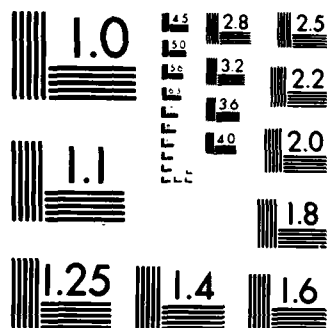
5/5

UNCLASSIFIED

F/G 9/2

NL

END



MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS 1963-A

FUNCTION 'FIRSTCH'

PURPOSE

FIND FIRST NON-BLANK IN CHARACTER VARIABLE

FUNCTIONAL CATEGORIES: M5

LANGUAGE: FORTRAN 77 EXTENDED

COMPUTERS (OPERATING SYSTEMS)
CDC 6000/CYBER 170 (NOS/BE)

REMARKS

NONE

USAGE

FIRSTCH (A, NCHAR)

DESCRIPTION OF PARAMETERS

A - CHARACTER STRING TO BE SCANNED
NCHAR - NUMBER OF CHARACTERS IN 'A' TO BE PROCESSED
(ACTUAL NUMBER PROCESSED IS THE LESSER OF 'NCHAR'
AND THE LENGTH OF 'A')
FIRSTCH - WILL CONTAIN THE POSITION OF THE FIRST NON-BLANK
(IF ALL BLANKS, 0 (ZERO) IS RETURNED)

CM REQUIRED: 47B

SUBPROGRAMS REQUIRED

PART OF LANGUAGE
AND MOD
OTHERS
NONE

AUTHOR

DAVID V SOMMER - DTNSRDC CODE 1892.2

DATE WRITTEN: 02/23/82

DATE(S) REVISED

LOCATION OF DECKS

SOURCE
UPDATE LIBRARY ON 'MSS: NSRDC5P,UN=CSYS
OBJECT
EDITLIB USER LIBRARY: NSRDC5

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC5,,FIRSTCH,OUTPUT,MSACCES=<PASSWORD>.

FUNCTION 'FRSTCH'

PURPOSE

FIND FIRST NON-BLANK IN CHARACTER VARIABLE

FUNCTIONAL CATEGORIES: M5

LANGUAGE: FORTRAN 77 EXTENDED

COMPUTERS (OPERATING SYSTEMS)
CDC 6000/CYBER 170 (NOS/BE)

REMARKS

NONE

USAGE

FRSTCH (A)

DESCRIPTION OF PARAMETERS

A - CHARACTER STRING TO BE SCANNED
FRSTCH - WILL CONTAIN THE POSITION OF THE FIRST NON-BLANK
(IF ALL BLANKS, 0 (ZERO) IS RETURNED)

CM REQUIRED: 43B

SUBPROGRAMS REQUIRED

PART OF LANGUAGE
AND MOD
OTHERS
NONE

AUTHOR

DAVID V SOMMER - DTNSRDC CODE 1892.2

DATE WRITTEN: 01/19/84

DATE(S) REVISED

LOCATION OF DECKS

SOURCE
UPDATE LIBRARY ON MSS: NSRDC5P,UN=CSYS
OBJECT
EDITLIB USER LIBRARY: NSRDC5

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC5,,FRSTCH,OUTPUT,MSACCES=<PASSWORD>.

FUNCTION 'GETSTR'

PURPOSE

EXTRACT CHARACTER STRING ACCORDING TO USER-DEFINED CRITERIA

FUNCTIONAL CATEGORIES: M5 M4

LANGUAGE: FORTRAN 77

COMPUTERS (OPERATING SYSTEMS)

CDC 6000/CYBER 170 (NOS/BE)

REMARKS

THE LENGTH OF THE EXTRACTED STRING IS THE SMALLEST OF:

- 1) THE NUMBER OF CHARACTERS WHICH MEET THE CRITERIA
- 2) THE IMPLIED LENGTH OF THE INPUT STRING
- 3) THE IMPLIED LENGTH OF THE OUTPUT STRING

IF NECESSARY, THE EXTRACTED STRING IS PADDED ON THE RIGHT WITH BLANKS.

USAGE

INTEGER GETSTR

...

N = GETSTR (IN, OUT, CODE, MATCH)

DESCRIPTION OF PARAMETERS

IN - CH** - IN - THE INPUT STRING

OUT - CH** - OUT - THE OUTPUT STRING

(FOR CODE=6/-6, USE ' ')

CODE - INT - IN - EXTRACTION CRITERIA. ONE OF:

1 - ALPHANUMERIC (LETTERS AND NUMBERS ONLY)

-1 - ALPHANUMERIC AND BLANK

2 - ALPHABETIC ONLY

-2 - ALPHABETIC AND BLANK

3 - NUMERIC ONLY

-3 - NUMERIC AND BLANK

4 - NUMERIC AND MINUS ('-')

-4 - NUMERIC AND MINUS AND BLANK

5 - WHILE IN <MATCH>

-5 - WHILE NOT IN <MATCH>

6 - SKIP WHILE IN <MATCH>

-6 - SKIP WHILE NOT IN <MATCH>

MATCH - CH** - IN - STRING OF ACCEPTABLE CHARACTERS

(FOR CODE=5/6)

STRING OF UNACCEPTABLE CHARACTERS

(FOR CODE=-5/-6)

(NOTE: THIS PARAMETER IS USED ONLY

CODE=5/-5/6/-6 AND MAY BE

OMITTED ON CDC. HOWEVER, TO

COMPLY WITH THE ANSI FORTRAN
77 STANDARD, IT SHOULD BE
SPECIFIED FOR THE OTHER VALUES
OF CODE. ' ' WILL MEET THE
REQUIREMENT.)

GETSTR - INT - OUT - WILL CONTAIN THE LENGTH OF THE
EXTRACTED OR SKIPPED STRING OR:

0 - NO STRING
-1 - CODE WAS INVALID

CM REQUIRED: 761B

EXAMPLES

- 1) EXTRACT 3 STRINGS FROM A "RECORD". THE FIRST STRING IS ALPHANUMERIC (7 CHARACTERS MAX), THE SECOND IS NUMERIC AND '-' (3 CHARACTERS MAX), THE THIRD IS EVERYTHING LEFT UP TO THE NEXT COMMA, BLANK, PERIOD OR RIGHT PARENTHESIS.

CHARACTER RECORD*80, FIRST*7, SECOND*20, THIRD*80

```
...
NEXT = 1
N1 = GETSTR (RECORD(NEXT:), FIRST, 1, ' ')
NEXT = NEXT + N1
N2 = GETSTR (RECORD(NEXT:) SECOND(1:3), 4, ' ')
NEXT = NEXT + N2
N3 = GETSTR (RECORD(NEXT:), THIRD, -5, ', .)')
```

- 2) AS EXAMPLE 1, EXCEPT SKIP LEADING BLANKS FOR EACH FIELD.

CHARACTER RECORD*80, FIRST*7, SECOND*20, THIRD*80

```
...
NEXT = 1
NEXT = NEXT + GETSTR (RECORD(NEXT:), ' ', 6, ' ')
N1 = GETSTR (RECORD(NEXT:), FIRST, 1, ' ')
NEXT = NEXT + N1
NEXT = NEXT + GETSTR (RECORD(NEXT:), ' ', 6, ' ')
N2 = GETSTR (RECORD(NEXT:), SECOND(1:3), 4, ' ')
NEXT = NEXT + N2
NEXT = NEXT + GETSTR (RECORD(NEXT:), ' ', 6, ' ')
N3 = GETSTR (RECORD(NEXT:), , THIRD, -5, ', .)')
```

- 3) EXTRACT 5 COMMA-SEPARATED PARAMETERS. NOTE THAT THE LAST PARAMETERS ENDS WITH A BLANK INSTEAD OF A COMMA.

CHARACTER*80 RECORD, STR1, STR2, STR3, STR4, STR5

```
...
NEXT = 1
N1 = GETSTR (RECORD(NEXT:), STR1, -5, ',')
NEXT = NEXT + N1 + 1
N2 = GETSTR (RECORD(NEXT:), STR2, -5, ',')
...
N5 = GETSTR (RECORD(NEXT:), STR5, -5, ' ')
```

SUBPROGRAMS REQUIRED

PART OF LANGUAGE

LEN MIN

OTHERS

NONE

AUTHOR

DAVID V SOMMER - DTNSRDC CODE 1892.2

DATE WRITTEN: 07/12/82

DATE(S) REVISED

LOCATION OF DECKS

SOURCE

UPDATE LIBRARY ON MSS: NSRDC5P,UN=CSYS

OBJECT

EDITLIB USER LIBRARY: NSRDC5

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC5,,GETSTR,OUTPUT,MSACCES=<PASSWORD>.

FUNCTION 'HMS2S'

PURPOSE

CONVERT HH.MM.SS TO SECONDS

FUNCTIONAL CATEGORIES: M2

LANGUAGE: FORTRAN 77

COMPUTERS (OPERATING SYSTEMS)
CDC 6000/CYBER 170 (NOS/BE)

REMARKS

NONE

USAGE

HMS2S (HMS)

DESCRIPTION OF PARAMETERS

HMS - CH** - TIME TO BE CONVERTED
(MAY BE 'HH.MM.SS. ', ' HH.MM.SS. ',
OR ' HH.MM.SS')

CM REQUIRED: 100B

SUBPROGRAMS REQUIRED

PART OF LANGUAGE

NONE

OTHERS

NONE

AUTHOR

DAVID V SOMMER - DTNSRDC CODE 1892.2

DATE WRITTEN: 05/01/74 (ISEC)

DATE(S) REVISED

03/18/83 - CONVERT TO FORTRAN 77
- CHANGE NAME FROM ISEC TO HMS2S

LOCATION OF DECKS

SOURCE

UPDATE LIBRARY ON MSS: NSRDC5P,UN=CSYS

OBJECT

EDITLIB USER LIBRARY: NSRDC5

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC5,,HMS2S,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'IDID'

PURPOSE

GET USER INITIALS AND INTERCOM USER ID FROM CHARGE CARD OR
LOGIN

FUNCTIONAL CATEGORIES: SO

LANGUAGE: FORTRAN 77 EXTENDED

COMPUTERS (OPERATING SYSTEMS)
CDC 6000/CYBER 170 (NOS/BE)

REMARKS

IF USER INITIALS AND USER ID ARE EQUAL, IT IS A BATCH JOB.

THE FTN4 VERSION HAS A SLIGHTLY DIFFERENT CALLING SEQUENCE.

USAGE

CALL IDID (ID, USERID)

DESCRIPTION OF PARAMETERS

ID - CH*4 - WILL CONTAIN 4-CHARACTER USER INITIALS
FROM CHARGE CARD OR START OF LOGIN
USERID - CH*10 - WILL CONTAIN 4-CHARACTER USER INITIALS
FROM CHARGE CARD OR UP TO 10-CHARACTER
USER ID FROM LOGIN
(IF ID = IUSERID, IT IS A BATCH JOB)

CM REQUIRED: 60B

METHOD

THE ID IS TAKEN FROM THE CONTROL POINT AREA.

SUBPROGRAMS REQUIRED

PART OF LANGUAGE

NONE

OTHERS

RCPA - READ CONTROL POINT AREA

AUTHOR

DAVID V SOMMER - DTNSRDC CODE 1892.2

DATE WRITTEN: 01/28/77

DATE(S) REVISED

12/10/81 - CONVERT TO FORTRAN 77 EXTENDED

LOCATION OF DECKS

SOURCE

UPDATE LIBRARY ON MSS: NSRDC5P, UN=CSYS

OBJECT

EDITLIB USER LIBRARY: NSRDC5

ANOTHER COPY OF THIS DOCUMENT

BEGIN, DOCGET, , NSRDC5, , IDID, OUTPUT, MSACCES=<PASSWORD>.

FUNCTION 'ITRANS'

PURPOSE

TRANSLATE CHARACTERS ACCORDING TO USER-SPECIFIED TRANSLATE TABLES

FUNCTIONAL CATEGORIES: M4

LANGUAGE: FORTRAN 77

COMPUTERS (OPERATING SYSTEMS)
CDC 6000/CYBER 170 (NOS/BE)

REMARKS

IF THE 'FROM' AND 'TO' TABLES DO NOT HAVE THE SAME LENGTH,
NO TRANSLATION OCCURS.

SEE 'TRANS' FOR THE SUBROUTINE VERSION.

USAGE

ITRANS (STRING, FROM, TO)

DESCRIPTION OF PARAMETERS

STRING - CH** - STRING WHOSE CHARACTERS ARE TO BE TRANSLATED
FROM - CH** - STRING OF CHARACTERS TO BE TRANSLATED
TO - CH** - STRING OF TRANSLATION CHARACTERS
ITRANS - INT - WILL CONTAIN ONE OF:
+N - THE NUMBER OF CHARACTERS TRANSLATED
0 - NO TRANSLATIONS
-1 - THE FROM AND TO TABLES HAVE DIFFERENT
LENGTHS - NO TRANSLATION OCCURRED

CM REQUIRED: 105B

EXAMPLES

IN ALL THE EXAMPLES, 'STRING' HAS BEEN DEFINED AS A
CHARACTER VARIABLE OF SOME LENGTH.

- 1) TRANSLATE ALL '_' IN A STRING TO '-':
NTR = ITRANS (STRING, '_', '-')
NTR WILL CONTAIN THE NUMBER OF CHARACTERS WHICH WERE
TRANSLATED
- 2) ADD 1 TO ALL DIGITS IN A STRING (9 BECOMES 0):
NTR = ITRANS (STRING, '0123456789', '1234567890')
- 3) CHANGE THE CHARACTERS ABCDE TO EDCBA, RESPECTIVELY:
NTR = ITRANS (STRING, 'ABCDE', 'EDCBA')
- 4) ILLUSTRATE AN INVALID CALL:
NTR = ITRANS (STRING, '1234', '123456')
ON RETURN, N WILL BE -1, BECAUSE THE TWO TRANSLATE
STRINGS ARE OF DIFFERENT LENGTHS.

SUBPROGRAMS REQUIRED
PART OF LANGUAGE
LEN
OTHERS
NONE

AUTHOR
DAVID V SOMMER - DTNSRDC CODE 1892.2

DATE WRITTEN: 07/25/83

DATE(S) REVISED

LOCATION OF DECKS
SOURCE DECK
UPDATE LIBRARY ON MSS: NSRDC5P,UN=CSYS
OBJECT DECK
EDITLIB USER LIBRARY: NSRDC5

ANOTHER COPY OF THIS DOCUMENT
BEGIN,DOCGET,,NSRDC5,,ITRANS,OUTPUT,MSACCES=<PASSWORD>.

FUNCTION 'JOBORG'

PURPOSE

DETERMINE JOB ORIGIN

FUNCTIONAL CATEGORIES: S0

LANGUAGE: FORTRAN 77 EXTENDED

COMPUTERS (OPERATING SYSTEMS)
CDC 6000/CYBER 170 (NOS/BE)

REMARKS

THE CALLING SEQUENCE DIFFERS FROM THE FORTRAN IV VERSION.

USAGE

JOBORG (ORG)

DESCRIPTION OF PARAMETERS

JOBORG - WILL CONTAIN ONE OF THE FOLLOWING:

- 1 - IF CALLING JOB IS A BATCH JOB
- 2 - FOR REAL TIME JOB
- 3 - FOR GRAPHICS JOB
- 4 - FOR MULTI-USER JOB
- 5 - FOR INTERCOM

ORG - WILL CONTAIN: 'BATCH', 'REAL TIME', 'GRAPHICS',
'MULTI-USER', OR 'INTERCOM', ACCORDING TO THE VALUE
OF 'JOBORG'.

CM REQUIRED: 54B

METHOD

THE INFORMATION IS TAKEN FROM THE CONTROL POINT AREA.

SUBPROGRAMS REQUIRED

PART OF LANGUAGE

AND

OTHERS

RCPA - READ CONTROL POINT AREA

AUTHOR

DAVID V SOMMER - DTNSRDC CODE 1892.2

DATE WRITTEN: 03/07/77

DATE(S) REVISED

09/30/82 - CONVERT TO FORTRAN 77

LOCATION OF DECKS

SOURCE

UPDATE LIBRARY ON MSS: NSRDC5P, UN=CSYS

OBJECT

EDITLIB USER LIBRARY: NSRDC5

ANOTHER COPY OF THIS DOCUMENT

BEGIN, DOCGET, , NSRDC5, , JOBORG, OUTPUT, MSACCES=<PASSWORD>.

FUNCTION 'LASTCH'

PURPOSE

DETERMINE NUMBER OF CHARACTERS THRU LAST NON-BLANK

FUNCTIONAL CATEGORIES: M5

LANGUAGE: FORTRAN 77 EXTENDED

COMPUTERS (OPERATING SYSTEMS)

CDC 6000/CYBER 170 (NOS/BE)

REMARKS

NONE

USAGE

LASTCH (A, NCHAR)

DESCRIPTION OF PARAMETERS

A - CHARACTER STRING TO BE SCANNED

NCHAR - NUMBER OF CHARACTERS IN 'A' TO BE PROCESSED
(ACTUAL NUMBER PROCESSED IS THE LESSER OF 'NCHAR'
AND THE LENGTH OF 'A')

LASTCH - WILL CONTAIN THE NUMBER OF CHARACTERS IN 'A'
EXCLUDING TRAILING BLANKS

CM REQUIRED: 47B

SUBPROGRAMS REQUIRED

PART OF LANGUAGE

AND MOD

OTHERS

NONE

AUTHOR

DAVID V SOMMER - DTNSRDC CODE 1892.2

DATE WRITTEN: 02/13/79

DATE(S) REVISED

07/15/81 - CONVERT TO FTN5

LOCATION OF DECKS

SOURCE

UPDATE LIBRARY ON MSS: NSRDC5P,UN=CSYS

OBJECT

EDITLIB USER LIBRARY: NSRDC5

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC5,,LASTCH,OUTPUT,MSACCES=<PASSWORD>.

FUNCTION 'LASTCHH'

PURPOSE

DETERMINE NUMBER OF CHARACTERS THRU LAST NON-BLANK IN A
HOLLERITH WORD OR ARRAY

FUNCTIONAL CATEGORIES: M5

LANGUAGE: FORTRAN 77 EXTENDED

COMPUTERS (OPERATING SYSTEMS)
CDC 6000/CYBER 170 (NOS/BE)

REMARKS

THE WORD IN 'A' WHICH CONTAINS THE LAST NON-BLANK CHARACTER
IS (LASTCHH(A,N)+9)/10.

THIS IS IDENTICAL TO FUNCTION 'LASTCH' FOR FTN4. THE FTN5
VERSION OF 'LASTCH' IS FOR CHARACTER DATA; 'LASTCHH' IS FOR
HOLLERITH DATA.

USAGE

LASTCHH (A, NCHAR)

DESCRIPTION OF PARAMETERS

A - ARRAY TO BE SCANNED
NCHAR - NUMBER OF CHARACTERS IN 'A' TO BE PROCESSED
LASTCHH - WILL CONTAIN THE NUMBER OF CHARACTERS IN 'A'
EXCLUDING TRAILING BLANKS

CM REQUIRED: 113B

SUBPROGRAMS REQUIRED

PART OF LANGUAGE
AND MOD
OTHERS
NONE

AUTHOR

DAVID V SOMMER - DTNSRDC CODE 1892.2

DATE WRITTEN: 02/13/79

DATE(S) REVISED

06/21/82 - NAME CHANGED TO LASTCHH TO ALLOW CO-EXISTENCE
WITH LASTCH (FOR CHARACTER DATA)
- CONVERTED TO FORTRAN 77

LOCATION OF DECKS

SOURCE
UPDATE LIBRARY ON MSS: NSRDC5P,UN=CSYS
OBJECT
EDITLIB USER LIBRARY: NSRDC5

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC5,,LASTCHH,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'LEFT'

PURPOSE

LEFT-JUSTIFY A CHARACTER STRING

FUNCTIONAL CATEGORIES: M4

LANGUAGE: FORTRAN 77 EXTENDED

COMPUTERS (OPERATING SYSTEMS)

CDC 6000/CYBER 170 (NOS/BE)

REMARKS

SEE ALSO CENTER, RIGHT.

USAGE

CALL LEFT (CH, WORK)

DESCRIPTION OF PARAMETERS

CH - CHARACTER STRING TO BE LEFT-JUSTIFIED IN PLACE

WORK - CHARACTER VARIABLE THE SAME LENGTH AS 'CH'

CM REQUIRED: 63B

EXAMPLES

CHARACTER*20 A, WORK

DATA A/ ' ABCDEFGHIJ '/

...

CALL LEFT (A, WORK)

'A' NOW CONTAINS 'ABCDEFGHIJ'

SUBPROGRAMS REQUIRED

PART OF LANGUAGE

LEN

OTHERS

FIRSTCH - FIND FIRST NON-BLANK CHARACTER

AUTHOR

DAVID V SOMMER - DTNSRDC CODE 1892.2

DATE WRITTEN: 07/15/82

DATE(S) REVISED

LOCATION OF DECKS

SOURCE

UPDATE LIBRARY ON MSS: NSRDC5P, UN=CSYS

OBJECT

EDITLIB USER LIBRARY: NSRDC5

ANOTHER COPY OF THIS DOCUMENT

BEGIN, DOCGET, , NSRDC5, , LEFT, OUTPUT, MSACCES=<PASSWORD>.

SUBROUTINE 'LGDATM'

PURPOSE

GET LOGIN DATE AND TIME

FUNCTIONAL CATEGORIES: Q0

LANGUAGE: FORTRAN 77

COMPUTERS (OPERATING SYSTEMS)
CDC 6000/CYBER 170 (NOS/BE)

REMARKS

NONE

USAGE

CALL LGDATM (LOGDAT, LOGTIM)

DESCRIPTION OF PARAMETERS

LOGDAT - CH*5 - WILL CONTAIN THE LOGIN DATE (YYJJJ)

LOGTIM - CH*4 - WILL CONTAIN THE LOGIN TIME (HHMM)

CM REQUIRED: 102

SUBPROGRAMS REQUIRED

PART OF LANGUAGE

AND MASK

OTHERS

CVHOCH - CONVERT HOLLERITH TO CHARACTER

REQTBL - READ INTERCOM USER TABLE

ARITHMETIC STATEMENT FUNCTIONS

GETBITS - EXTRACT BITS FROM A WORD

AUTHOR

DAVID V. SOMMER - DTNSRDC CODE 1892.2

DATE WRITTEN: 10/05/84

DATE(S) REVISED

LOCATION OF DECKS

SOURCE DECK

UPDATE LIBRARY ON MSS: NSRDC5P,UN=CSYS

OBJECT DECK

EDITLIB USER LIBRARY: NSRDC5

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC5,,LGDATM,OUTPUT,MSACCES=<PASSWORD>.

FUNCTION 'LSTCH'

PURPOSE

DETERMINE NUMBER OF CHARACTERS THRU LAST NON-BLANK

FUNCTIONAL CATEGORIES: M5

LANGUAGE: FORTRAN 77 EXTENDED

COMPUTERS (OPERATING SYSTEMS)
CDC 6000/CYBER 170 (NOS/BE)

REMARKS
NONE

USAGE
LSTCH (A)

DESCRIPTION OF PARAMETERS

A - CHARACTER STRING TO BE SCANNED
LSTCH - WILL CONTAIN THE NUMBER OF CHARACTERS IN 'A'
EXCLUDING TRAILING BLANKS

CM REQUIRED: 45B

SUBPROGRAMS REQUIRED
PART OF LANGUAGE
AND MOD
OTHERS
NONE

AUTHOR
DAVID V SOMMER - DTNSRDC CODE 1892.2

DATE WRITTEN: 01/19/84

DATE(S) REVISED

LOCATION OF DECKS
SOURCE

UPDATE LIBRARY ON MSS: NSRDC5P,UN=CSYS
OBJECT
EDITLIB USER LIBRARY: NSRDC5

ANOTHER COPY OF THIS DOCUMENT
BEGIN,DOCGET,,NSRDC5,,LSTCH,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'MFRAME'

PURPOSE

OBTAIN THE MACHINE AND MAINFRAME RUNNING THE PROGRAM

FUNCTIONAL CATEGORIES: Q0

LANGUAGE: FORTRAN 77 EXTENDED

COMPUTERS (OPERATING SYSTEMS)

CDC 6000/CYBER 170 (NOS/BE)

REMARKS

NONE

USAGE

CALL MFRAME (CPU, MF)

DESCRIPTION OF PARAMETER

CPU - WILL RETURN MACHINE ON WHICH THE PROGRAM IS RUNNING

(CDC FTN4: INTEGER, LEFT-ADJ, BLANK-FILLED;

CDC FTN5: CHARACTER*6)

(WILL RETURN ONE OF:

'6700', '6600', '6400', 'CY74', 'CY750', 'CY176',
'CY825')

MF - WILL RETURN MAINFRAME ON WHICH THE PROGRAM IS RUNNING

(CDC FTN4: INTEGER, LEFT-ADJ, BLANK-FILLED;

CDC FTN5: CHARACTER*3)

(WILL RETURN ONE OF:

'MFA', 'MFB', 'MFC', 'MFD', 'MFE', 'MFF',
'MFG', 'MFZ')

CM REQUIRED: 320B

SUBPROGRAMS REQUIRED

PART OF LANGUAGE

AND SHIFT

OTHERS

MACHINE - GET SYSTEM MACHINE INFORMATION

ARITHMETIC STATEMENT FUNCTIONS

R38FMT - FAST R-FORMAT DECODE (RIGHT-ADJ, ZERO-FILLED)

AUTHOR

DAVID V SOMMER - DTNSRDC CODE 1892.2

DATE WRITTEN: 03/15/79

DATE(S) REVISED

08/15/80 - ADD "CY74" FOR CYBER 74
05/05/81 - CONVERT TO FTN5 ON CDC
07/01/82 - UPGRADE TO LEVEL 552
 - ADD SUPPORT FOR MFE, MFF, MFG
08/18/83 - CHANGE TEST OF CPU FOR MFG

LOCATION OF DECKS

SOURCE

UPDATE LIBRARY ON MSS: NSRDC5P,UN=CSYS

OBJECT

EDITLIB USER LIBRARY: NSRDC5

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC5,,MFRAME,OUTPUT,MSACCES=<PASSWORD>.

FUNCTION 'MF2CPU'

PURPOSE

RETURN CPU NAME CORRESPONDING TO SUPPLIED MAINFRAME NAME

FUNCTIONAL CATEGORIES: 20

LANGUAGE: FORTRAN 77

COMPUTERS (OPERATING SYSTEMS)

CDC 6000/CYBER 170 (NOS/BE)

REMARKS

NONE

USAGE

CHARACTER*6 CPU, MF2CPU

CHARACTER*3 MF

CPU = MF2CPU (MF)

DESCRIPTION OF PARAMETERS

MF - CH*3 - INPUT MAINFRAME NAME OR LETTER
(E.G., 'MFF' OR 'F')

MF2CPU - CH*6 - WILL RETURN THE CORRESPONDING CPU NAME
(E.G., 'CY176')
-OR- ' ', IF MF IS NOT RECOGNIZED

CM REQUIRED: 321B

SUBPROGRAMS REQUIRED

PART OF LANGUAGE

NONE

OTHERS

NONE

AUTHOR

DAVID V SOMMER - DTNSRDC CODE 1892.2

DATE WRITTEN: 04/20/82

DATE(S) REVISED

04/20/82 - CONVERT TO FORTRAN 77

LOCATION OF DECKS

SOURCE

UPDATE LIBRARY ON MSS: NSRDC5P,UN=CSYS

OBJECT

EDITLIB USER LIBRARY: NSRDC5

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC5,,MF2CPU,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'NEWDAT'

PURPOSE

ADD/SUBTRACT SPECIFIED NUMBER OF DAYS TO/FROM A GIVEN DATE

FUNCTIONAL CATEGORIES: M2

LANGUAGE: FORTRAN 77 EXTENDED

COMPUTERS (OPERATING SYSTEMS)

CDC 6000/CYBER 170 (NOS/BE)

REMARKS

NONE

USAGE

CALL NEWDAT (FMT, OLD, NEW, OCENT, NCENT, ADD)

DESCRIPTION OF PARAMETERS

FMT - FORMAT OF DATES (INTEGER)

1 - 'MM/DD/YY '

2 - ' MM/DD/YY '

3 - 'MMDDYY'

-1 - 'YY/MM/DD '

-2 - ' YY/MM/DD '

-3 - 'YYMMDD'

OLD - OLD DATE (CHARACTER * 10)

NEW - NEW DATE (CHARACTER * 10)

OCENT - OLD CENTURY (E.G., INTEGER 1900)

NCENT - NEW CENTURY (E.G., INTEGER 1900)

ADD - INGETER NUMBER OF DAYS TO ADD
(NEGATIVE TO SUBTRACT)

CM REQUIRED: 372B

SUBPROGRAMS REQUIRED

PART OF LANGUAGE

NONE

OTHERS

JGDATE - JULIAN/GREGORIAN DATE CONVERTER (MULTI-YEAR)

AUTHOR

DAVID V SOMMER - DTNSRDC CODE 1892.2

DATE WRITTEN: 1968

DATE(S) REVISED

02/ /73 - CONVERT TO SCOPE 3.3

07/09/79 - CONVERT TO BURROUGHS B7700

05/06/81 - CONVERT TO CDC FORTRAN 77

LOCATION OF DECKS

SOURCE

UPDATE LIBRARY ON MSS: NSRDC5P,UN=CSYS

OBJECT

EDITLIB USER LIBRARIES: NSRDC5

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC5,,NEWDAT,OUTPUT,MSACCES=<PASSWORD>.

FUNCTION 'NUMER'

PURPOSE

TEST STRING FOR NUMERICS

FUNCTIONAL CATEGORIES: M5

LANGUAGE: FORTRAN 77

COMPUTERS (OPERATING SYSTEMS)

CDC 6000/CYBER 170 (NOS/BE)

REMARKS

NONE

USAGE

NUMER (STRING)

DESCRIPTION OF PARAMETERS

STRING - CHARACTER STRING TO BE TESTED

NUMER - WILL CONTAIN ONE OF:

0 - THE STRING CONTAINS ONLY DIGITS

+N - THE POSITION OF THE FIRST NON-DIGIT

CM REQUIRED: 51B

SUBPROGRAMS REQUIRED

PART OF LANGUAGE

INDEX LEN

OTHERS

NONE

ARITHMETIC STATEMENT FUNCTIONS

DIGIT - TEST CHARACTER FOR DIGIT

AUTHOR

DAVID V SOMMER - DTNSRDC CODE 1892.2

DATE WRITTEN: 07/13/83

DATE(S) REVISED

LOCATION OF DECKS

SOURCE DECK

UPDATE LIBRARY ON MSS: NSRDC5P,UN=CSYS

OBJECT DECK

EDITLIB USER LIBRARY: NSRDC5

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC5,,NUMER,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'PFRC'

PURPOSE

SUPPLY DESCRIPTION OF PERMANENT FILE FUNCTION RETURN CODE

FUNCTIONAL CATEGORIES: Q0

LANGUAGE: FORTRAN 77 EXTENDED

COMPUTERS (OPERATING SYSTEMS)
CDC 6000/CYBER 170 (NOS/BE)

REMARKS

THE DESCRIPTIONS ARE THOSE FOUND IN THE "NOS/BE VERSION 1
REFERENCE MANUAL" (60493800 L) ON PAGE 7-84.

USAGE

CALL PFRC (RC, MSG)

DESCRIPTION OF PARAMETERS

RC - INT - RETURN CODE FROM THE PERMANENT FILE
FUNCTION
MSG - CH*50 - WILL CONTAIN THE DESCRIPTION OF THE
SUPPLIED 'IRC'
(IF 'RC' IS INVALID, 'UNKNOWN RETURN CODE'
IS RETURNED)

CM REQUIRED: 1027B

SUBPROGRAMS REQUIRED

PART OF LANGUAGE

NONE

OTHERS

NONE

AUTHOR

DAVID V SOMMER - DTNSRDC CODE 1892.2

DATE WRITTEN: 05/18/76

DATE(S) REVISED

02/14/77 - UPDATE FOR NOS/BE 1.0

07/15/80 - UPDATE FOR NOS/BE 1.4 (LEVEL 508)

02/02/82 - CONVERT TO FORTRAN 77

LOCATION OF DECKS

SOURCE

UPDATE LIBRARY ON MSS: NSRDC5P,UN=CSYS

OBJECT

EDITLIB USER LIBRARY: NSRDC5

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC5,,PFRC,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'PM'

PURPOSE

WRITE 'PM' PRINTER MESSAGE

FUNCTIONAL CATEGORIES: Q0

LANGUAGE: FORTRAN 77

COMPUTERS (OPERATING SYSTEMS)
CDC 6000/CYBER 170 (NOS/BE)

REMARKS

NONE

USAGE

CALL PM (OUTFYL, FC)

DESCRIPTION OF PARAMETERS

OUTFYL - INT - OUTPUT FILE UNIT NUMBER -OR-
 -1 FOR SYSTEM OUTPUT FILE
FC - CHAR - 2-CHARACTER FORMS CODE -OR-
 1- TO 30-CHARACTER MESSAGE DESCRIBING
 PRINTER CHANGES REQUIRED -OR-
 ' ' FOR THE MESSAGE 'RESTORE PRINTER'

CM REQUIRED: 206B

OUTPUT DESCRIPTION

ONE OF THE FOLLOWING LINES IS WRITTEN TO THE SPECIFIED FILE:

'1T'
 'PM 1-PART NARROW UNLINED PAPER'

'<FC>'
 'PM MOUNT SPECIAL FORMS <FC>'

'<UP-TO-30-CHARACTER MESSAGE>'
 'PM<UP-TO-30-CHARACTER MESSAGE>'

' '
 'PM RESTORE PRINTER'

OUTPUT UNITS

UNIT #	LFN	USE
-----	-----	-----
USER SPECIFIED		LISTABLE OUTPUT (WILL CONTAIN PM LINE)

EXAMPLE

USE ONE-PART NARROW PAPER WRITING TO SYSTEM OUTPUT FILE:

```
...  
EXTERNAL EOJ  
...  
CALL PM (-1, '1T')  
CALL RECOVR (EOJ, 0"77", 0)  
...  
CALL PM (-1, ' ' )  
END  
SUBROUTINE EOJ  
...  
CALL PM (-1, ' ' )  
...  
RETURN  
END
```

THE FIRST CALL TO PM WRITES THE MESSAGE 'PM 1-PART NARROW UNLINED PAPER'. THE 'EXTERNAL OJ' AND 'CALL EOJ ...' ARE NEEDED IN CASE THE PROGRAM SHOULD TERMINATE ABNORMALLY. IT MUST BE SURE THAT THE 'PM RESTORE PRINTER' MESSAGE IS WRITTEN, IF NOT BY THE SECOND CALL ABOVE, THEN BY THE REPRIEVE ROUTINE 'EOJ'. NOTICE THAT THE SECOND CALL IN THIS EXAMPLE IS THE LAST EXECUTABLE STATEMENT IN THE PROGRAM. THIS IS TO PREVENT A SECOND 'RESTORE PRINTER' MESSAGE IN CASE THE PROGRAM TERMINATES ABNORMALLY AFTER THE FIRST ONE WAS WRITTEN.

SUBPROGRAMS REQUIRED

PART OF LANGUAGE
LEN
OTHERS
NONE

AUTHOR

DAVID V SOMMER - DTNSRDC CODE 1892.2

DATE WRITTEN: 07/15/83

DATE(S) REVISED

LOCATION OF DECKS

SOURCE DECK
UPDATE LIBRARY ON MSS: NSRDC5P,UN=CSYS
OBJECT DECK
EDITLIB USER LIBRARY: NSRDC5

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC5,,PM,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'PRTYM'

PURPOSE

GET AND PRINT CPA, CPB, CP, PP, IO AND WALL CLOCK TIMES
SINCE LAST CALL AND PRINT USER-SUPPLIED MESSAGE

FUNCTIONAL CATEGORIES: Q4 J4

LANGUAGE: FORTRAN 77

COMPUTERS (OPERATING SYSTEMS)
CDC 6000/CYBER 170 (NOS/BE)

REMARKS
NONE

USAGE

CALL PRTYM (OUTFYL, TIMES, CLOK, MSG)
CALL PRTYM (OUTFYL, TIMES, CLOK, '0')

DESCRIPTION OF PARAMETERS

OUTFYL - INT - OUTPUT UNIT FOR PRINTED LINE
(FORTRAN LOGICAL UNIT NUMBER (1-999))
(FOR STANDARD OUTPUT FILE (I.E., PRINT OR
WRITE (*,)), USE -1)
TIMES - REAL - 6-WORD ARRAY TO CONTAIN THE FOLLOWING:
1 - ELAPSED CPA TIME IN SECONDS
2 - ELAPSED CPB TIME IN SECONDS
3 - ELAPSED CP TIME IN SECONDS (CPA+CPB)
4 - ELAPSED PP TIME IN SECONDS
5 - ELAPSED IO TIME IN SECONDS
6 - ELAPSED WALL CLOCK TIME IN SECONDS
CLOK - CH*10 - ELAPSED WALL CLOCK TIME (HH.MM.SS.)
MSG - CH** - MESSAGE TO BE PRINTED
(IF MSG IS '0', HEADINGS, BUT NOT TIMES,
WILL BE PRINTED.)

CM REQUIRED: 146B

OUTPUT UNITS

UNIT #	LFN	USE

USER SPECIFIES...	LISTABLE OUTPUT	

EXAMPLE

```
      PROGRAM TEST (OUTPUT=128, ...  
      CHARACTER CLOK*10  
      REAL TIMES(6)  
C     GET INITIAL TIMES AND PRINT HEADING  
      CALL PRTYM (-1, TIMES, CLOK, '0')  
      .....  
C     GET ELAPSED TIMES AND PRINT WITH MESSAGE  
      CALL PRTYM (-1, TIMES, CLOK, 'TEST NUMBER 1')  
      ...  
C     NEW HEADINGS ARE NOT NEEDED, SO CALL ELTYM DIRECTLY  
      CALL ELTYM (TIMES, CLOK)  
      .....  
C     GET ELAPSED TIMES AND PRINT WITH MESSAGE  
      CALL PRTYM (-1, TIMES, CLOK, 'TEST NUMBER 2')  
      ...  
      END
```

SUBPROGRAMS REQUIRED

PART OF LANGUAGE

NONE

OTHERS

ELTYM - GET ELAPSED TIME SINCE LAST CALL

AUTHOR

DAVID V SOMMER - DTNSRDC CODE 1892.2

DATE WRITTEN: 04/20/76 (PRTIME)

DATE(S) REVISED

03/18/83 - CONVERT TO FORTRAN 77

- CHANGE NAME FROM PRTIME TO PRTYM

LOCATION OF DECKS

SOURCE

UPDATE LIBRARY ON MSS: NSRDC5P,UN=CSYS

OBJECT

EDITLIB USER LIBRARY: NSRDC5

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC5,,PRTYM,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'RIGHT'

PURPOSE

RIGHT-JUSTIFY A CHARACTER STRING

FUNCTIONAL CATEGORIES: M4

LANGUAGE: FORTRAN 77 EXTENDED

COMPUTERS (OPERATING SYSTEMS)
CDC 6000/CYBER 170 (NOS/BE)

REMARKS

SEE ALSO CENTER, LEFT.

USAGE

CALL RIGHT (CH, WORK)

DESCRIPTION OF PARAMETERS

CH - CHARACTER STRING TO BE RIGHT-JUSTIFIED IN PLACE
WORK - CHARACTER VARIABLE THE SAME LENGTH AS 'CH'

CM REQUIRED: 104B

EXAMPLES

CHARACTER*20 A, WORK
DATA A/ ' ABCDEFGHIJ '/
...
CALL RIGHT (A, WORK)

'A' NOW CONTAINS ' ABCDEFGHIJ'

SUBPROGRAMS REQUIRED

PART OF LANGUAGE
LEN
OTHERS
LASTCH - FIND LAST NON-BLANK CHARACTER

AUTHOR

DAVID V SOMMER - DTNSRDC CODE 1892.2

DATE WRITTEN: 07/15/82

DATE(S) REVISED

LOCATION OF DECKS

SOURCE
UPDATE LIBRARY ON MSS: NSRDC5P, UN=CSYS
OBJECT
EDITLIB USER LIBRARY: NSRDC5

ANOTHER COPY OF THIS DOCUMENT

BEGIN, DOCGET, , NSRDC5, , RIGHT, OUTPUT, MSACCES=<PASSWORD>.

SUBROUTINE 'ROUTERC'

PURPOSE

SUPPLY DESCRIPTION OF ROUTE RETURN CODE

FUNCTIONAL CATEGORIES: Q0

LANGUAGE: FORTRAN 77 EXTENDED

COMPUTERS (OPERATING SYSTEMS)

CDC 6000/CYBER 170 (NOS/BE)

REMARKS

THE DESCRIPTIONS ARE THOSE FOUND IN THE "NOS/BE VERSION 1
REFERENCE MANUAL" (60493800 M) ON PAGE 7-81.

USAGE

CALL ROUTERC (IRC, TEXT)

DESCRIPTION OF PARAMETERS

IRC - INT - RETURN CODE FROM SUBROUTINE 'LF ("ROUTE", '
TEXT - CH*50 - WILL CONTAIN THE DESCRIPTION OF THE
SUPPLIED 'IRC'
(IF 'IRC' IS INVALID, 'UNKNOWN RETURN CODE'
IS RETURNED)

CM REQUIRED: 605B

SUBPROGRAMS REQUIRED

PART OF LANGUAGE

NONE

OTHERS

NONE

AUTHOR

DAVID V SOMMER - DTNSRDC CODE 1892.2

DATE WRITTEN: 12/15/77

DATE(S) REVISED

05/17/83 - CONVERT TO FORTRAN 77

LOCATION OF DECKS

SOURCE

UPDATE LIBRARY ON MSS: NSRDC5P,UN=CSYS

OBJECT

EDITLIB USER LIBRARY: NSRDC5

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC5,,ROUTERC,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'SETREW'

PURPOSE

CONVERT REWIND OPTION INTO OPEN AND CLOSE CODES

LANGUAGE: FORTRAN 77

COMPUTERS (OPERATING SYSTEMS)
CDC 6000/CYBER 170 (NOS/BE)

REMARKS
NONE

USAGE

CALL SETREW (REW, OPEN, CLOSE, NOE)

DESCRIPTION OF PARAMETERS

REW - CH*2 - INPUT REWIND OPTION. ONE OF:
A - OPEN=NOREWIND; CLOSE=REWIND
B - OPEN=REWIND ; CLOSE=NOREWIND
E - OPEN=POSITION BEFORE EOI;
CLOSE=NOREWIND
EN - OPEN=POSITION BEFORE EOI;
CLOSE=NOREWIND
ER - OPEN=POSITION BEFORE EOI;
CLOSE=REWIND
EU - OPEN=POSITION BEFORE EOI;
CLOSE=UNLOAD
R - OPEN=REWIND; CLOSE=REWIND
U - OPEN=REWIND; CLOSE=REWIND AND UNLOAD
OTHER - OPEN=NOREWIND; CLOSE=NOREWIND
(ANY WORDS BEGINNING WITH THESE LETTERS WILL
PRODUCE THE SAME RESULTS.)
OPEN - CH*1 - WILL CONTAIN OPEN REWIND OPTION
('E', 'N', 'R')
CLOSE - CH*1 - WILL CONTAIN CLOSE REWIND OPTION
('N', 'R', 'U')
NOE - INT - 0 - ALLOW ALL VALUES OF REW
OTHER - DO NOT ALLOW 'E' VALUES OF REW

CM REQUIRED: 321B

SUBPROGRAMS REQUIRED

PART OF LANGUAGE

NONE

OTHERS

NONE

AUTHOR

DAVID V SOMMER - DTNSRDC CODE 1892.2

DATE WRITTEN: 10/29/75

DATE(S) REVISED

01/29/76

01/11/76 - ADD 'NOE' PARAMETER

09/28/82 - CONVERT TO FORTRAN 77

LOCATION OF DECKS

SOURCE

UPDATE LIBRARY ON MSS: NSRDC5P,UN=CSYS

OBJECT

EDITLIB USER LIBRARY: NSRDC5

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC5,,SETREW,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'SM5PRNT'

PURPOSE

PRINT CONTENTS OF SORT/MERGE 5 STATISTICS ARRAY

FUNCTIONAL CATEGORIES: NO

LANGUAGE: FORTRAN 77

COMPUTERS (OPERATING SYSTEMS)

CDC 6000/CYBER 170 (NOS/BE)

REMARKS

NONE

USAGE

CALL SM5PRNT (OUTFYL, SM5ARRAY)

DESCRIPTION OF PARAMETERS

OUTFYL - OUTPUT UNIT NUMBER (1-999) -OR-

-1 TO INDICATE STANDARD OUTPUT FILE

SM5ARRAY - ARRAY USED IN CALL TO SM5SORT OR SM5MERG

CM REQUIRED: 154B

OUTPUT DESCRIPTION

THE FOLLOWING IS PRINTED ON THE SPECIFIED FILE AFTER A BLANK
LINE:

NNNNNNNNNN RECORDS READ
NNNNNNNNNN RECORDS DELETED BY OWN1
NNNNNNNNNN RECORDS INSERTED BY OWN1
NNNNNNNNNN RECORDS INSERTED BY OWN2
NNNNNNNNNN RECORDS SORTED
NNNNNNNNNN RECORDS DELETED BY PWN3
NNNNNNNNNN RECORDS INSERTED BY OWN3
NNNNNNNNNN RECORDS INSERTED BY OWN4
NNNNNNNNNN <RESERVED>
NNNNNNNNNN RECORDS DELETED BY OWN5
NNNNNNNNNN RECORDS COMBINED FOR SUMMING
NNNNNNNNNN RECORDS WRITTEN
MINIMUM RECORD LENGTH NNNNNNNNNN CHARACTERS
AVERAGE RECORD LENGTH NNNNNNNNNN CHARACTERS
MAXIMUM RECORD LENGTH NNNNNNNNNN CHARACTERS

OUTPUT UNITS

UNIT	#	LFN	USE
------	---	-----	-----

<USER SPECIFIED>			SORT SUMMARY
------------------	--	--	--------------

SUBPROGRAMS REQUIRED

PART OF LANGUAGE

NONE

OTHERS

NONE

AUTHOR

DAVID V SOMMER - DTNSRDC CODE 1892.2

DATE WRITTEN: 04/83

DATE(S) REVISED

LOCATION OF DECKS

SOURCE DECK

UPDATE LIBRARY ON MSS: NSRDC5P,UN=CSYS

OBJECT DECK

EDITLIB USER LIBRARY: NSRDC5

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC5,,SM5PRNT,OUTPUT,MSACCES=<PASSWORD>.

FUNCTION 'S2HMS'

PURPOSE

CONVERT SECONDS TO ' HH.MM.SS. '

FUNCTIONAL CATEGORIES: M2

LANGUAGE: FORTRAN 77

COMPUTERS (OPERATING SYSTEMS)

CDC 6000/CYBER 170 (NOS/BE)

REMARKS

NONE

USAGE

S2HMS (SEC)

DESCRIPTION OF PARAMETERS

SEC - INT - TIME (IN SECONDS) TO BE CONVERTED

S2HMS - CH*10 - WILL CONTAIN ' HH.MM.SS. '

SUBPROGRAMS REQUIRED

PART OF LANGUAGE

NONE

OTHERS

NONE

CM REQUIRED: 46B

AUTHOR

DAVID V SOMMER - DTNSRDC CODE 1892.2

DATE WRITTEN: 05/08/74 (IHMS)

DATE(S) REVISED

03/18/83 - CONVERT TO FORTRAN 77

- CHANGE NAME FROM IHMS TO S2HMS

LOCATION OF DECKS

SOURCE

UPDATE LIBRARY ON MSS: NSRDC5P,UN=CSYS

OBJECT

EDITLIB USER LIBRARY: NSRDC5

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC5,,S2HMS,OUTPUT,MSACCES=<PASSWORD>.

FUNCTION 'TRANS'

PURPOSE

TRANSLATE CHARACTERS ACCORDING TO USER-SPECIFIED TRANSLATE TABLES

FUNCTIONAL CATEGORIES: M4

LANGUAGE: FORTRAN IV

COMPUTERS (OPERATING SYSTEMS)
CDC 6000/CYBER 170 (NOS/BE)

REMARKS

IF THE 'FROM' AND 'TO' TABLES DO NOT HAVE THE SAME LENGTH,
NO TRANSLATION OCCURS.

SEE 'ITRANS' FOR THE FUNCTION VERSION WHICH WILL ALSO RETURN
THE NUMBER OF CHARACTERS TRANSLATED.

USAGE

CALL TRANS (STRING, FROM, TO)

DESCRIPTION OF PARAMETERS

STRING - CH** - STRING WHOSE CHARACTERS ARE TO BE TO BE
TRANSLATED
FROM - CH** - STRING OF CHARACTERS TO BE TRANSLATED
TO - CH** - STRING OF TRANSLATION CHARACTERS

CM REQUIRED: 77B

EXAMPLES

IN ALL THE EXAMPLES, 'STRING' HAS BEEN DEFINED AS A
CHARACTER VARIABLE OF SOME LENGTH.

- 1) TRANSLATE ALL '_' IN A STRING TO '-':
CALL TRANS (STRING, '_', '-')
- 2) ADD 1 TO ALL DIGITS IN A STRING (9 BECOMES 0):
CALL TRANS (STRING, '0123456789', '1234567890')
- 3) CHANGE THE CHARACTERS ABCDE TO EDCBA, RESPECTIVELY:
CALL TRANS (STRING, 'ABCDE', 'EDCBA')
- 4) ILLUSTRATE AN INVALID CALL:
CALL TRANS (STRING, '1234', '123456')
NO TRANSLATION OCCURS BECAUSE THE TWO TRANSLATE TABLES
HAVE DIFFERENT LENGTHS.

SUBPROGRAMS REQUIRED
PART OF LANGUAGE
LEN
OTHERS
NONE

AUTHOR
DAVID V SOMMER - DTNSRDC CODE 1892.2

DATE WRITTEN: 07/25/83

DATE(S) REVISED

LOCATION OF DECKS
SOURCE DECK
UPDATE LIBRARY ON MSS: NSRDC5P,UN=CSYS
OBJECT DECK
EDITLIB USER LIBRARY: NSRDC5

ANOTHER COPY OF THIS DOCUMENT
BEGIN,DOCGET,,NSRDC5,,TRANS,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'TTYMSG'

PURPOSE

DRIVER TO WRITE A LINE TO AN INTERACTIVE TERMINAL

FUNCTIONAL CATEGORIES: L0 J4

LANGUAGE: FORTRAN 77

COMPUTERS (OPERATING SYSTEMS)

CDC 6000/CYBER 170 (NOS/BE)

REMARKS

THE SUBROUTINE INCLUDES AN OPTIONAL ENTRY TO OPEN THE TTY
FILE SPECIFYING A UNIT NUMBER OTHER THAN THE DEFAULT OF 77.
IT ALSO INCLUDES AN ENTRY TO CLOSE AND RETURN THE FILE.

USAGE

CALL TTYOPEN (UNIT)

CALL TTYMSG (MSG)

CALL TTYCLOS

DESCRIPTION OF PARAMETERS

UNIT - FORTRAN UNIT NUMBER TO BE USED FOR THE TTY FILE
(IF THE CALL TTYOPEN IS OMITTED, UNIT 77 IS USED)
MSG - CHARACTER MESSAGE TO BE WRITTEN

CM REQUIRED: 103B

OUTPUT UNITS

UNIT #	LFN	USE
77	YYTTY	CONNECTED TTY FILE (OPENED BY TTYOPEN, WHICH MAY CHANGE THE UNIT NUMBER, OR BY THE FIRST CALL TO TTYMSG - CLOSED AND RETURNED BY TTYCLOS)

EXAMPLE

CHARACTER*30 NAME

...
CALL TTYOPEN (66)

...
CALL TTYMSG ('ENTER YOUR NAME-')
READ (66, '(A)') NAME

...
CALL TTYCLOS

SUBPROGRAMS REQUIRED
PART OF LANGUAGE
CONNEC
OTHERS
NONE

AUTHOR
DAVID V SOMMER - DTNSRDC CODE 1892.2

DATE WRITTEN: 07/22/82

DATE(S) REVISED

LOCATION OF DECKS
SOURCE
UPDATE LIBRARY ON MSS: NSRDC5P,UN=CSYS
OBJECT
EDITLIB USER LIBRARY: NSRDC5

ANOTHER COPY OF THIS DOCUMENT
BEGIN,DOCGET,,NSRDC5,,TTYMSG,OUTPUT,MSACCES=<PASSWORD>.

SUBROUTINE 'TTYOPN'
SUBROUTINE 'TTYCLO'

PURPOSE

OPEN INTERACTIVE INPUT AND OUTPUT FILES

FUNCTIONAL CATEGORIES: Q3

LANGUAGE: FORTRAN 77

COMPUTERS (OPERATING SYSTEMS)
CDC 6000/CYBER 170 (NOS/BE)

REMARKS

NONE

USAGE

CALL TTYOPN (IN, OUT) << OPEN AND CONNECT FILES IN AND
OUT
CALL TTYCLO << CLOSE AND RETURN FILES

DESCRIPTION OF PARAMETERS

IN - INPUT UNIT NUMBER
OUT - OUTPUT UNIT NUMBER

CM REQUIRED: 66B

INPUT/OUTPUT UNITS

UNIT #	LFN	USE
USER	CONSOL	CONNECTED OUTPUT FILE
SPECIFIES		
USER	KEYBRD	CONNECTED INPUT FILE
SPECIFIES		

EXAMPLE

CALL TTYOPN (1, 2)
...
CALL TTYCLO

SUBPROGRAMS REQUIRED

PART OF LANGUAGE
CONNECT
OTHERS
NONE

AUTHOR

DAVID V SOMMER - DTNSRDC CODE 1892.2

DATE WRITTEN: 06/21/83

DATE(S) REVISED

LOCATION OF DECKS

SOURCE DECK

UPDATE LIBRARY ON MSS: NSRDC5P,UN=CSYS

OBJECT DECK

EDITLIB USER LIBRARY: NSRDC5

ANOTHER COPY OF THIS DOCUMENT

BEGIN,DOCGET,,NSRDC5,,TTYOPN,OUTPUT,MSACCES=<PASSWORD>.

INITIAL DISTRIBUTION

COPIES:

12 DIRECTOR
DEFENSE TECHNICAL INFORMATION CENTER (DTIC)
CAMERON STATION
ALEXANDRIA, VIRGINIA 23314

CENTER DISTRIBUTION

COPIES:

1	18/1809	GLEISSNER, G. H.
1	1804	AVRUNIN, L.
1	1805	CUTHILL, E. H.
2	1808	WILDY, D.
1	182	CAMARA, A. W.
1	184	SCHOT, J. W.
1	185	CORIN, T.
1	187	ZUBKOFF, M. J.
1	189	GRAY, G. R.
1	189.2	HAYDEN, H. P.
1	189.3	MORRIS, JEAN
40	1892.1	STRICKLAND, J. D.
5	1892.2	SOMMER, D. V.
1	1892.3	MINOR, L. R.
1	1894	SEALS, W.
1	1896	GLOVER, A.
1	1896.2	DENNIS, L.
1	522	TIC (C)
1	522.2	TID (A)
1	93	PATENT COUNSEL

DTNSRDC ISSUES THREE TYPES OF REPORTS

1. DTNSRDC REPORTS, A FORMAL SERIES, CONTAIN INFORMATION OF PERMANENT TECHNICAL VALUE. THEY CARRY A CONSECUTIVE NUMERICAL IDENTIFICATION REGARDLESS OF THEIR CLASSIFICATION OR THE ORIGINATING DEPARTMENT.
2. DEPARTMENTAL REPORTS, A SEMIFORMAL SERIES, CONTAIN INFORMATION OF A PRELIMINARY, TEMPORARY, OR PROPRIETARY NATURE OR OF LIMITED INTEREST OR SIGNIFICANCE. THEY CARRY A DEPARTMENTAL ALPHANUMERICAL IDENTIFICATION.
3. TECHNICAL MEMORANDA, AN INFORMAL SERIES, CONTAIN TECHNICAL DOCUMENTATION OF LIMITED USE AND INTEREST. THEY ARE PRIMARILY WORKING PAPERS INTENDED FOR INTERNAL USE. THEY CARRY AN IDENTIFYING NUMBER WHICH INDICATES THEIR TYPE AND THE NUMERICAL CODE OF THE ORIGINATING DEPARTMENT. ANY DISTRIBUTION OUTSIDE DTNSRDC MUST BE APPROVED BY THE HEAD OF THE ORIGINATING DEPARTMENT ON A CASE-BY-CASE BASIS.

END

FILMED

1-85

DTIC